S Y S T E M   I N D E X   A N D   D E S C R I P T I O N

3B20S PROCESSOR

JULY, 1981

PRELIMINARY COPY

SYSTEM INDEX AND DESCRIPTION

3B20S PROCESSOR

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

CONTENTS PAGE

1. INTRODUCTION

The system description is a broad overview of the 3B20S Processor. It provides a general description of the hardware, software, features, and functions of the 3B20S Processor system.

The system index provides a list of the 3B20S Processor documents and includes a description of the contents of each. The index also relates the documents to system software (generic) releases. The index will be a separate document in the future.

The general section provides a basic description of the overall system including the 3B20S Processor capabilities, internal interfaces, and external inputs and outputs.

The hardware section provides a physical description of the 3B20S Processor cabinets, units, and circuit packs.

The functional section describes the functions and operations of the individual 3B20S Processor units. Internal functions as well as interactions with other units are included.

The feature section provides a description of the system features including control unit, peripheral unit, power, packaging, diagnostic, and software features.

The software sections include descriptions of the UNIX (trademark of Bell Telephone Laboratories) operating system and UNIX kernel utilized by the 3B20S Processor.

NOTES

BASIC DOCUMENTS INDEX

	SYSTEM GENERIC RELEASES				
	4.1	x	y	z	
Identification	**Issue**				**Title**
254-301-900WE	P1	-	-	-	System Index*
254-301-910WE	P1	-	-	-	System Index and Description
254-301-920WE	P1	-	-	-	UNIX User's Guide
254-301-925WE	P1	-	-	-	UNIX User's Manual
254-301-926WE	P1	-	-	-	UNIX Administrator's Manual
254-301-930WE	P1	-	-	-	System Administrator's Guide
254-301-940WE	P1	-	-	-	System Operations Guide
254-301-950WE	P1	-	-	-	System Maintenance Guide

*The system index is currently combined with the system description but will be issued as a separate document in the future.

SUPPLEMENTAL DOCUMENTS INDEX

	SYSTEM GENERIC RELEASES				
	4.1	x	y	z	
Identification	**Issue**				**Title**
254-301-970WE	P1	-	-	-	Detailed Hardware Descriptions
254-301-980WE	P1	-	-	-	Detailed Software Descriptions
254-301-990WE	P1	-	-	-	Remedial (Trouble Clearing) Strategies

2. SYSTEM INDEX

GENERAL

The index provides information that relates 3B20S Processor documents to system software (generic) releases and provides the unique document identification (control) number, issue, and title for each document.

Identification Number - Each document is identified with a unique identification number.

Issue - The issue number of a document.

Title - A unique but short descriptive phrase given each document and by which the document may be known.

System Generic Release - The number that identifies each supported generic release of the software that supports a system.

A release is assigned only when there is a new release or a changed release of the software supporting the system.

The 3B20S Processor System documentation is divided into two types of documentation: (1) basic documentation and (2) supplemental documentation. The basic documentation (one set) is supplied with the purchase of a system. The supplemental documentation is not supplied with the purchase of a system but is available to be purchased as desired by the users.

The basic documentation provides a high-level description of the system and provides information on how to utilize the basic features of the system. High-level preventive (routine) support information and vendor manuals for the supplied vendor (KS) equipment are also provided.

The supplemental documentation provides detailed descriptive information, theory, and remedial (trouble clearing) support information, and also provides descriptive/operational information about special features (in addition to the basic features) offered with the 3B20S Processor.

BASIC DOCUMENTATION

Basic documentation is user-oriented and provides the information and procedures necessary for the user to understand the hardware and software functions sufficiently to utilize the features of the 3B20S Processor. The basic documentation provides high-level descriptive, operational, and support service

information. The basic documentation (one set, including vendor documentation supplied with KS equipment) is provided with the purchase of a 3B20S Processor. Additional copies of the documents may be purchased through the customer sales account representative. Sufficient information is provided for the 3B20S Processor user to operate the processor, to understand the UNIX operating system, and to perform preventive (routine) support procedures.

The 3B20S Processor basic documentation includes the following:

Title	Description
System Index and Description 254-301-910WE	Provides a brief description of the 3B20S Processor documents, identifies each document by title and identification number, and relates each document issue to associated software generic releases.
Index	Provides information that relates the 3B20S Processor documents to the system software (generic) releases and provides the unique document identification (control) numbers, issue, and title for the documents.
Introduction	Provides a general overview of the system to aid in understanding the system hardware and software features and the system functions.
Hardware Description	Describes system space and environmental requirements and the cabinets, cabinet units, interfaces, and power requirements.
Functional Description	Provides functional description of the processor units, peripheral control units, peripheral devices, cooling units, and power.
System Features	Describes features for the hardware and software, including the control unit, peripheral devices, diagnostic programs, power, and software/firmware.
Software Description	Provides a general description of the UNIX operating system and the software features provided with the 3B20S Processor.
UNIX Kernel	Describes the UNIX kernel, the file system, and its management.

Title	Description
UNIX User's Guide 254-301-920WE	Describes the following subjects and provides descriptive and procedural information on how to use the 3B20S Processor and the UNIX operating system.
General	Provides a general description of the system capabilities available to the UNIX user, user interfaces, and the user environment.
System Functions	Describes features and their purpose, typical terminal printouts/displays precautions and constraints.
UNIX User's Manual 254-301-925WE	Describes the commands, system calls, subroutines, special files, file formats, library functions, subroutines, and miscellaneous UNIX features. This volume contains the information needed by a typical UNIX user or programmer.
UNIX Administrator's Manual 254-301-926WE	Contains commands used to administer a system, a description of all the special files (device drivers) for the 3B20S Processor, and hardware oriented procedures and facilities descriptions. This volume complements the information in the UNIX User's Manual.
System Administrator's Guide 254-301-930WE	Provides descriptive and procedural information needed by the system administrator to coordinate and manage the 3B20S Processor. Identifies access procedures, administrative inputs/outputs, cautions, security, system protection, configuration changes, etc.
System Operations Guide 254-301-940WE	Provides descriptive and procedural information needed to keep the 3B20S Processor operational. Describes hardware/software operations (initializations, bootstrapping), process operations, file system procedures, and input/output messages and formats.

Title	Description
System Maintenance Guide 254-301-950WE	Describes how to perform limited support service, such as preventive (routine) maintenance, on the 3B20S Processor; how failures are detected and reported to operational or service personnel; and procedures to be followed to restore the system or escalate the trouble to higher levels of support services.
General	Describes the support service philosophy, strategies, and procedures used with the 3B20S Processor.
Preventive Maintenance	Describes the preventive (routine) support information used to perform preventive services and normal operational services on devices such as the tape transport and moving head disk. These services are normally performed on a scheduled basis.

SUPPLEMENTAL DOCUMENTATION

The supplemental documentation provides detailed (hardware and software) descriptive, theory, and trouble clearing strategies for users who require more in-depth information. Also included is descriptive, procedural, and tutorial information for special features not included as part of the basic 3B20S Processor documentation. Detailed descriptive, theory, and support service information is available to be purchased by those users who plan to support the 3B20S Processor themselves rather than subscribing to the Western Electric Company support services offering. This information is for users who desire more technical information to perform trouble clearing and support services or to better understand the 3B20S Processor and its capabilities and special features.

The 3B20S Processor supplemental documentation includes the following:

Title	Description
254-301-970WE	This volume contains documents that provide detailed descriptive/theory for the 3B20S Processor hardware. These descriptions provide information the reader will find useful in understanding the functional operation of the processor for trouble clearing and for understanding the functional operation of the processor.
254-301-980WE	Provides detailed software descriptions and other documents that are useful in understanding the functional operation of the software associated with the 3B20S Processor. Also includes information useful in using and understanding the software features of the processor.
254-301-990WE	Provides detailed support information and procedures for remedial (trouble clearing) strategies. This information is used to isolate troubles and direct remedial support activities, direct remedial actions, and restore services of the processor.

3. GENERAL INFORMATION

SECTION PURPOSE

In this section of the system description, the 3B20S Processor is described in general terms. The system configuration is provided along with system capabilities, internal machine interfaces, and external inputs and outputs.

SYSTEM CONFIGURATION

The 3B20S Processor is a high-speed, general purpose processor that can be used in a wide variety of applications because of high reliability and flexibility. The processor complex is an intergrated package of both hardware (the 3B20S Processor System) and software (the UNIX operating system).

A fixed cabinet lineup and connectorized cabling provide for quick installation of the 3B20S Processor. Optional growth of the 3B20S Processor can be easily accomplished by adding a growth cabinet and growth units to the minimum system configuration.

The minimum configuration of the 3B20S Processor (Fig. 3.1) requires approximately 170 square feet of floor space (including the recommended access space) and includes four cabinets and three peripheral units (see Section 4).

The system may be expanded by adding a growth frame, video terminals, printers, or moving head disk transports (maximum of eight for each disk file controller installed) to the minimum configuration. Figure 3.2 illustrates a typical growth configuration.

The power and input/output signal cabling between the 3B20S Processor cabinets is via port holes in the side of each cabinet. Cabling between the cabinets and the moving head disk transports, video terminals, and printers can be run overhead in vertical cabling ducts or below the floor if the 3B20S Processor is installed on a raised floor.

SYSTEM CAPABILITIES

A. Processor

The central processing unit of the 3B20S Processor is a self-checking 32-bit processor. Self-checking logic ensures that errors are detected when they occur instead of later when more drastic recovery measures would be required to correct the problem.

The 3B20S Processor can be equipped with a maximum of 8 megabytes of main store memory. The main store is growable in increments of 512 kilobytes. The main store memory controller performs error detection and correction. Hamming code circuitry in the controller provides correction of all single bit errors and the detection of all double errors. The controller also performs various hardware checks on internal circuitry and bus communications. When an error does occur, the central control unit is signaled with the appropriate error condition.

The processor is also equipped with a cache store unit. The cache store is a temporary memory that provides storage for frequently read main store locations. It provides the processor with 8 kilobytes of high-speed memory for instructions and data and 8 kilobytes for an interrupt stack. The high-speed access of the cache memory reduces the time required to process main store information.

B. Periphery

All of the 3B20S Processor peripherals have direct memory transfer capability with the main store memory. The central control unit builds job blocks for a peripheral and that peripheral performs the requested job and notifies the central control when the job is completed. From the time the central control completes building the job block until the peripheral unit transfers the desired data into or out of the main memory, the central control is free to perform other tasks.

Video Terminal

The video terminal provides a means of communication with the 3B20S Processor. The operator types commands on the terminal keyboard and status indications and return messages are displayed on the video terminal. The video terminal can display reversed video and flashing characters and has a split screen capability so that status messages and other long term information can be continuously displayed on part of the screen while other messages and displays may be placed on the other parts of the screen.

Line Printer

The 3B20S Processor line printer controller supports two parallel output, medium-speed, line printer ports. Each port on the controller can support a line printer with a rate of 2000 lines per minute, although the total combined speed of both printers cannot exceed 2000 lines per minute.

Moving Head Disk Transports

The 3B20S Processor is equipped with 300 megabyte moving head disk transports. Each disk file controller equipped in the 3B20S Processor is capable of interfacing the 3B20S Processor with up to eight disk transports. The 300 megabyte disk transport has a sequential power-up capability which reduces the power surge as

the motors of the 300 megabyte disk transports are started. When the first drive is up to speed, the next transport is started. This sequence continues until all disk transports in the system are started.

C. Power Backup

The 3B20S Processor is equipped with four 6-cell, 12-volt, maintenance-free batteries as a backup for the rectifier in the event of an ac power failure. These batteries will provide backup power to the 3B20S Processor circuitry for approximately 15 minutes.

INTERNAL MACHINE INTERFACES

The 3B20S Processor can be functionally divided into three sections:

- o Control Unit
- o Periphery (directly controlled by the control unit)
- o Power.

Figure 3.3 illustrates the interfaces between the control unit and the periphery.

A. Control Unit

The control unit is comprised of the following units which are located in the processor cabinet:

- o Central Control
- o Main Store
- o Direct Memory Access Unit
- o Input/Output Channels.

Central Control

The central control is the center of all the operations of the 3B20S Processor, and it interfaces with all of the other units of the control unit. Control information and address data is sent from central control to the direct memory access unit, input/output channels, and main store.

Main Store

The main store contains system program instructions and data. It can be accessed by central control or by the direct memory access unit.

Direct Memory Access Unit and Input/Output Channels

The 3B20S Processor may be equipped with a maximum of five programmable input/output channels and a maximum of four direct memory access channels. These channels are controlled directly by central control. The input/output channels provide an interface between the 3B20S Processor and application peripheral devices. The direct memory access channels provide the capability for direct data transfers between the main store and the directly controlled system periphery (disk file controller and input/output processor). This enables data to be transferred into and out of the main store without going through central control.

B. Periphery

The major peripheral units that are directly controlled by central control are the input/output processor (input/output peripheral control cabinet) and the disk file controller (disk controller/magnetic tape cabinet). Application peripheral devices are interfaced to central control via input/output channels or the input/output processor.

Input/Output Processor

Each input/output processor interfaces the direct memory access unit to as many as 16 peripheral controllers to enable the transfer of data between peripheral devices (e.g., video terminal, printer, etc.) and main store without requiring the use of central control. This relieves central control of many routine control and data transfers. Each peripheral controller may interface with a multiple of terminal devices.

Disk File Controller

The disk file controller provides microprocessor control to interpret and execute commands from central control to enable information transfers to and from the moving head disk transports. The disks are used to store maintenance data, application software, operating system software, and initialization data.

C. Power

The 3B20S Processor requires ac source voltages of 117 Vac single phase, 208 Vac three phase, and 208 Vac single phase. The 117 Vac is provided through the service outlets for units such as the video terminal, printer, test equipment, etc. The 208 Vac three phase is the source voltage for the rectifier that supplies the −48 Vdc which is distributed via buses to the 3B20S Processor cabinets. The 208 Vac single phase is required to operate the

moving head disk transports.

EXTERNAL INPUTS AND OUTPUTS

The major interfaces with the 3B20S Processor include:

- o Input/Output Channels
- o Input/Output Processor
- o Video terminal
- o Printer.

A. Input/Output Channels

The input/output channels are located in the processor cabinet and provide an interface between the 3B20S Processor and application peripheral devices. The 3B20S Processor may have a maximum of five input/output channels.

Dual serial channels are used for the input/output channels and each consists of one circuit pack. Dual serial channels are used for medium- and high-speed peripheral devices. Each dual serial channel may interface with a maximum of 16 peripheral devices. Serial data is divided into high and low bits and is transmitted simultaneously via two cables between the dual serial channel and a peripheral device.

B. Input/Output Processor

Each input/output processor basic unit can interface the 3B20S Processor control unit with as many as 32 peripheral devices. Each input/output growth unit can interface with an additional 32 peripheral devices. This enables the transfer of data between the devices and main store without requiring the direct involvement of the central control unit. Central control is therefore relieved of many routine control and data transfers thus enabling it to be better used as a system controller.

Each input/output processor (basic or growth) may be equipped with a maximum of eight peripheral controllers. There are two line communities with four peripheral controllers in each community. The peripheral controllers are the interface between the input/output processor and the peripheral devices. Each peripheral controller can interface with as many as four slow to medium-speed peripheral devices.

C. Video Terminal and Printer

The video terminal is the primary operator interface for the 3B20S Processor. Requests and commands are input on the terminal keyboard and status indications and return messages are displayed on the video terminal. Hard copy printouts of system responses

can be obtained on the associated printer.

SOFTWARE

A. General

The 3B20S Processor is a general purpose, multiuser processor which utilizes the UNIX operating system. The UNIX operating system provides a convenient working environment and a uniform set of tools for efficient development of computer programs, text and documentation preparation, and other user-defined applications. The UNIX operating system is a disk-oriented system featuring:

- o Hierarchical file systems utilizing demountable volumes
- o Multiprogramming
- o Device independence
- o Invisible access methods
- o Compatible file, device, and interprocess input/output
- o Ability to initiate asynchronous, sequential, and background processes
- o Context and full screen editor
- o Data protection mechanism based on user identification (ID) assigned to files
- o Many utility, languages, and application programs.

B. The UNIX Kernel

The UNIX kernel is the software on which everything else depends and is the only code that cannot be replaced by the user. The kernel maintains the file system, supports system calls, and manages system resources. This code always resides in memory. The following are functions supported by the kernel:

- o Process control
- o File system
- o Input/output (I/O) system
- o Device drivers
- o Libraries.

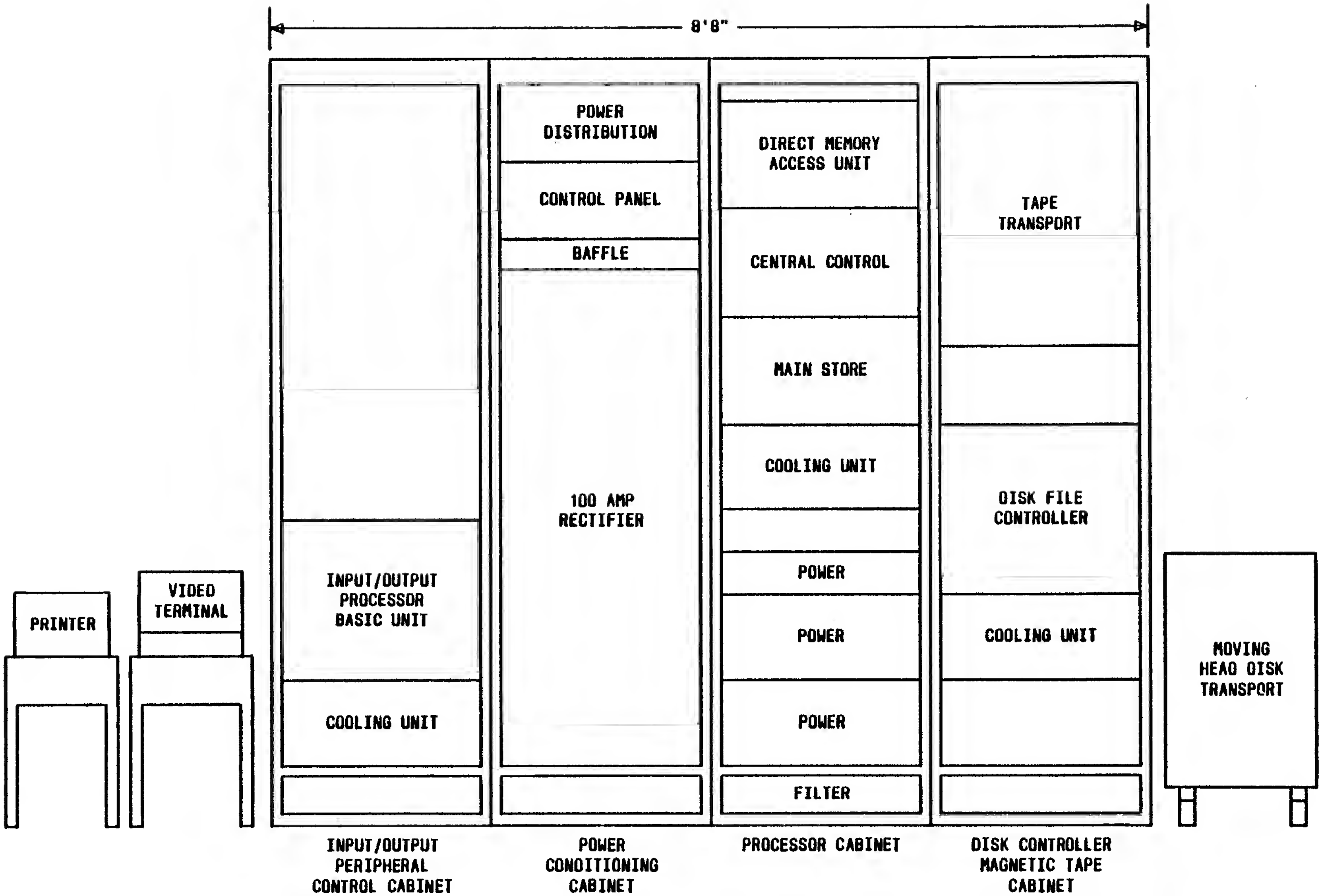

Fig. 3.1--3B20S Processor Minimum Configuration

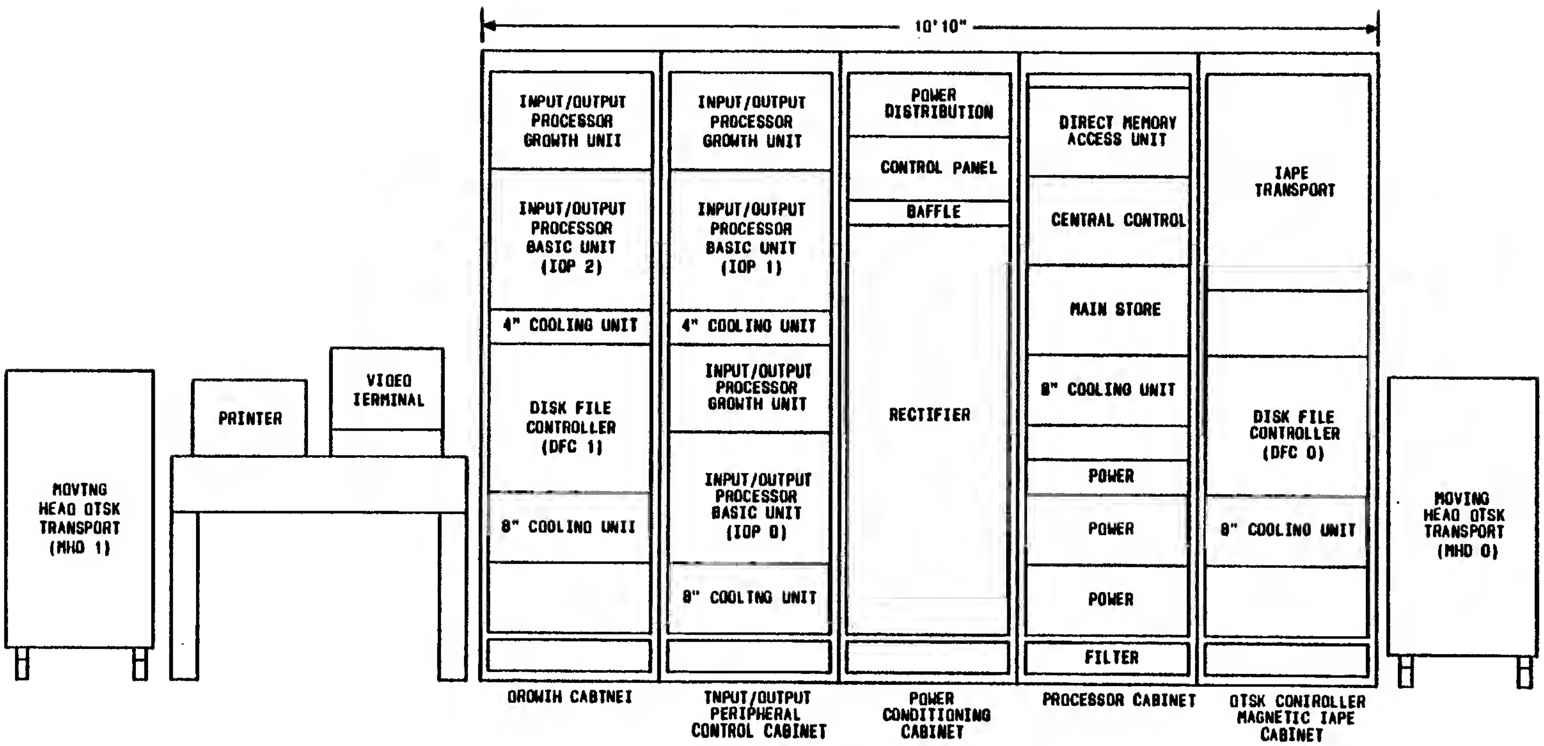

Fig. 3.2--3B20S Processor Typical Growth Configuration

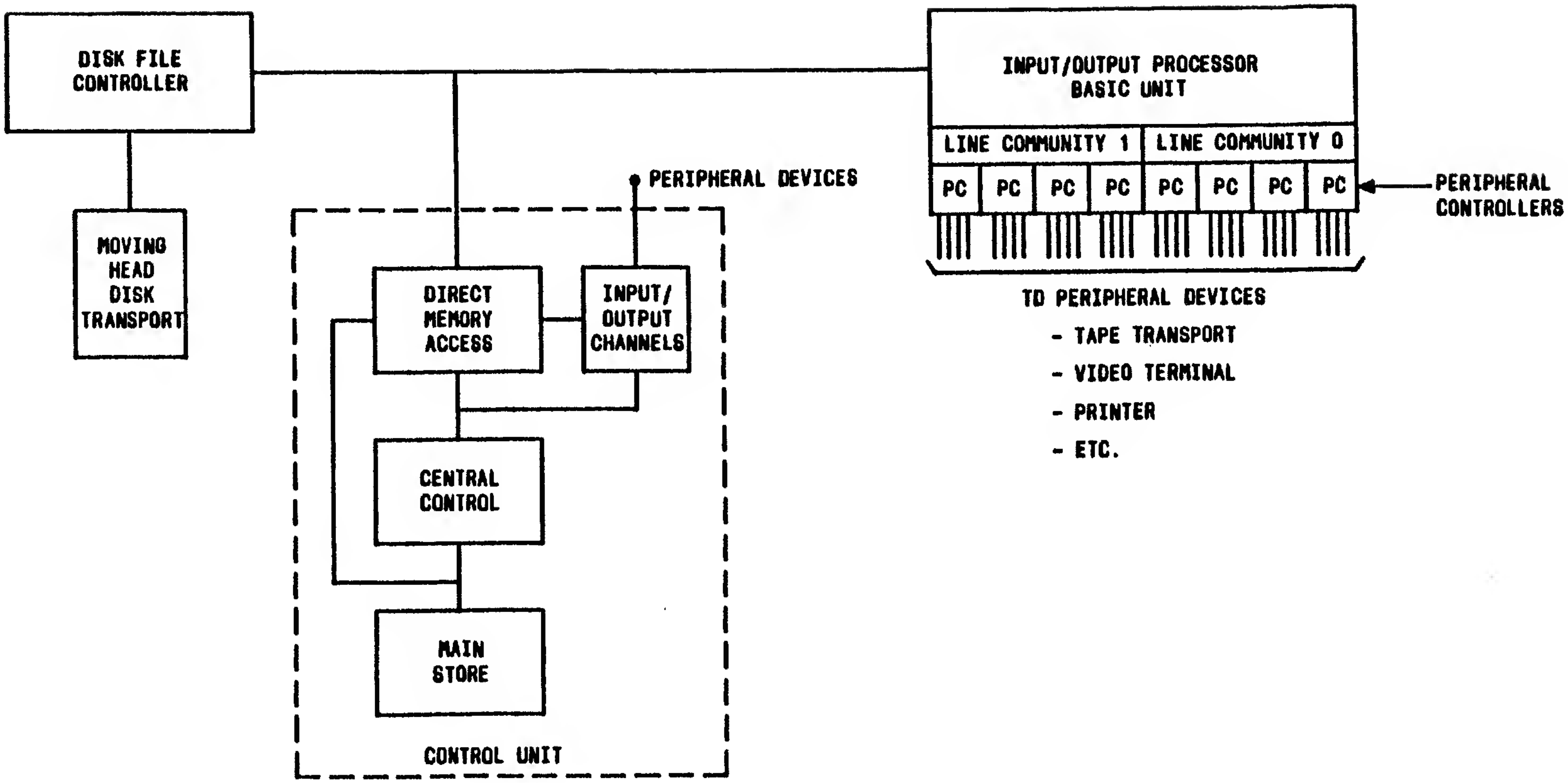

Fig. 3.3--3B20S Processor Basic Functional Diagram

NOTES

4. HARDWARE DESCRIPTION

SECTION PURPOSE

This section of the system description describes the hardware configuration of the 3B20S Processor. Included are physical descriptions of:

- o System cabinets
- o Cabinet units
- o Circuit packs
- o Power units
- o Peripheral units
- o System interfaces
- o Space and environment requirements.

CABINETS AND UNITS

The standard cabinet size is 2 feet 2 inches wide, 2 feet deep, and 6 feet high. A hinged door is on the front of the cabinet for access to circuit packs and other unit components such as fuses, manual controls, and cooling units where provided. Also, a hinged door is on the rear of the cabinet for access to the backplane wiring and cabling.

The cabinets may be set alone, mounted to the floor, or earthquake mounted depending on the needs of the user and local building code requirements. The video terminal and printer are mounted on pedestals. The moving head disk transports may be bolted to the floor or on rollers. If desired, the equipment can be mounted on a raised floor. In all cases, the cabinets must be electrically insulated from the floor.

The basic circuit pack in the 3B20S Processor measures 8 inches by 13 inches and is equipped with either a 200 pin connector (TN code) or a 300 pin connector (UN code). The circuit packages are standard transistor-transistor logic circuits and transistor-transistor logic compatible large-scale integration circuits. Low power intergrated circuits are used to minimize power consumption and heat dissipation.

The basic 3B20S Processor (Fig. 4.1) consists of the following cabinets and peripheral units:

- o Processor Cabinet

- o Input/Output Peripheral Control Cabinet
- o Disk Controller/Magnetic Tape Cabinet
- o Power Conditioning Cabinet
- o Moving Head Disk Transport
- o Video Terminal
- o Printer.

Table A lists the dimensions of the cabinets, cabinet units, and peripheral units.

A. Processor Cabinet

The processor cabinet (Fig. 4.2) consists of the following units:

- o Direct Memory Access Unit
- o Central Control
- o Main Store
- o Cooling Unit
- o Power Units.

Direct Memory Access Unit

The direct memory access unit transfers blocks of data directly between peripheral devices and the main store unit. The direct memory access unit is controlled by the central control unit.

The direct memory access unit consists of the following subunits (Fig. 4.3):

- o Direct Memory Access Controller--One or two are installed and each requires four circuit packs (two UN35, UN36, UN37).
- o Direct Memory Access Channels--Direct memory access controller 0 can be equipped with up to three dual serial channels. Direct memory access controller 1 can be equipped with only one dual serial channel. Each dual serial channel is provided on one circuit pack (UN9).
- o Input/Output Channels--Five input/output channel positions are provided and each channel can be equipped with one dual serial channel (UN9).

Central Control Unit

The central control unit provides the control functions for the 3B20S Processor and contains the following subunits (Fig. 4.3):

- o Microstore--Contains read-only memory (one circuit pack - UN28) and writable microstore (two circuit packs - UN48).

- o Central Processing Unit--Contains the microcontrol (UN44), data manipulation units (UN1B and UN23B), special registers (UN2 and UN3), store data controller (UN6), store address controller (UN43), and store address translator (UN45).

- o Main Store Update Unit--Used with the direct memory access unit to update the memory in the main store. One circuit pack is required (UN34).

- o Cache Store--Provides bypassing of the main store for frequently used information. Three circuit packs are required (cache controller - two UN10; cache memory - UN11).

Main Store Unit

The main store unit provides a storage area for information to be used by the 3B20S Processor when performing system functions. The main store responds to commands from the central control or the direct memory access unit.

The main store unit contains the following subunits (Fig. 4.3):

- o Power Switch (ABB1)--Provides manual control of the processor cabinet power. One circuit pack is required.

- o Emergency Action Interface (EAI)--Provides a means, via the maintenance terminal, of manually forcing certain 3B20S Processor recovery configurations in the event that automatic system error recovery is unsuccessful. Processor recovery data is returned to maintenance personnel at the maintenance terminal. One circuit pack is required (TN11).

- o Main Store Controller--Provides the interface between main store memory and the 3B20S Processor units. Two circuit packs are required (UN39 and UN40).

- o Main Store Array--Provides the storage for program instructions and information. Sixteen 0.5 megabyte circuit packs (TN14) are required for maximum capacity (8 megabytes). However, only two packs (1 megabyte) are

equipped in the minimum configuration.

Cooling Unit

The cooling unit provides for cooling, filtering of incoming air, and air circulation within the processor cabinet. The cooling unit is located directly below the main store unit. It is divided into three sections, each containing a fan, control circuitry, filter, and an alarm LED.

Power Units

The power units are located at the bottom of the processor cabinet and include:

- o Three 244D dc-to-dc converters (-48V to +5V) (A fourth converter can be added for maximum capacity.)
- o One J1C129AE power unit comprised of an ED-4C188-30 dc-to-dc converter (-48V to +5V, +12V and -12V) and a 132AJ dc-to-dc converter (-48V to -5V and +12V).

B. Input/Output Peripheral Control Cabinet

The input/output peripheral control cabinet (Fig. 4.4) consists of the following units:

- o Input/Output Processor Basic Unit
- o Cooling Unit.

This cabinet can be expanded to include another input/output processor basic unit, two input/output processor growth units, and a 4-inch cooling unit (Fig. 4.5).

The purpose of the input/output processor units is to provide:

- o An interface between peripheral devices and the 3B20S Processor
- o An autonomous input/output function via use of the direct memory access unit into the main store unit
- o Autonomous controls for the transfers of data blocks to and from peripheral devices
- o Buffering and data formatting as required by the peripheral devices.

Input/Output Processor Basic Unit

The input/output processor basic unit consists of the following subunits (Fig. 4.6):

- o Power Switch (ABB1)--Provides for manual power control of

the basic unit and associated growth unit. One circuit pack is required.

- Peripheral Interface Controller--Provides the interface between the direct memory access unit and the input/output processor basic and growth units. Also, it provides control functions for the line communities, manual power control for the input/output processor basic and growth unit, and power (136N dc-to-dc converter; -48V to +5V). The controller contains the duplex dual serial bus selector (TN69), bus interface controller (TN70), microcontrol store (TN84), peripheral interface controller (TN61), and input/output microprocessor interface (UN25).

- Line Community (two provided)--These function as the interface between the input/output processor basic unit and the peripheral device connected to the input/output processor basic unit via a peripheral controller. Each line community consists of up to four peripheral controller circuit packs, one power monitor circuit pack (TN71), and one 136N dc-to-dc converter.

Input/Output Processor Growth Unit

The input/output processor growth unit increases the capability of the input/output processor basic unit by the addition of two line communities. Each line community contains up to four peripheral controller circuit packs, one power monitor circuit pack (TN71), and one 136N dc-to-dc converter. This unit is controlled by the peripheral interface controller in the input/output processor basic unit.

Peripheral Controllers

The peripheral controllers provide the interface between the input/output processor units and the peripheral devices requiring the input/output processor functions. These controllers are plugged into the line communities (maximum of four controllers per community).

The peripheral controllers provided for the basic 3B20S Processor include:

- Nine-Track Tape Transport Controller (UN32)

- Synchronous Line Controller (data link) (TN75B)

- Asynchronous Maintenance Terminal Controller (TN74)

- Data Link Controller (TN82)

- Maintenance Video Terminal Controller (TN83).

Optional 3B20S Processor peripheral controllers include:

- o Additional Basic System Controllers
- o High-Speed Tape Controller (UN52)
- o Line Printer Controller (TN85)
- o Diagnostic Processor (UN38 and UN51).

Cooling Unit

The cooling units provide cooling, filter incoming air (8-inch only), and circulate air within the cabinet. A cooling unit is required for each input/output processor basic unit and is located directly below the unit. It is divided into three sections, each containing a fan, control circuitry, and an alarm LED. The 8-inch cooling unit also contains an air filter.

C. Disk Controller/Magnetic Tape Cabinet

The disk controller/magnetic tape cabinet (Fig. 4.7) consists of the following units:

- o Tape Transport
- o Disk File Controller
- o Cooling Unit.

Tape Transport

The KS-22091 tape transport is a 9-track recorder that can record and reproduce data in a 1600 bit-per-inch phase-encoded format.

Disk File Controller

The disk file controller provides the interface between the moving head disk transport and the 3B20S Processor. Each disk file controller is capable of interfacing with up to a maximum of eight moving head disk transports. The disk file controller also provides for control of the data transfers between the 3B20S Processor and the moving head disk transports.

The disk file controller consists of the following subunits (Fig. 4.8):

- o Power Switch (ABB1)--Provides for manual power control of the disk file controller. One circuit pack is required.
- o Peripheral Interface Controller--A microprocessor that controls all information transfers between the 3B20S Processor and the moving head disks. One circuit pack is required (TN61).
- o Microcontrol Store--Contains the instructions that the

peripheral interface controller executes to control information transfers. Three circuit packs are required (TN62).

- o Parallel Serial Data Interface--Performs data format conversion and error correction functions for transfers between the disk file controller and the moving head disks. One circuit pack is required (TN65).

- o Bus Interface Controller--Provides the interface between the duplex dual serial bus selector and the peripheral interface controller. One circuit pack is required (TN70).

- o Duplex Dual Serial Bus Selector--Provides the interface between the processor and the disk file controller. One circuit pack is required (TN69).

- o Moving Head Disk Control--Provides control data for the moving head disk transport operation. One circuit pack is required (TN63).

- o Moving Head Disk Data Clock--Provides for the internal serial data timing within the parallel serial data interface. One circuit pack (TN64) is required for the first eight moving head disk transports equipped. A second circuit pack (TN64) is required for the installation of moving head disk transports nine through sixteen.

- o Power Monitor--Provides for control of the disk file controller power sequencing. One circuit pack is required (TN73).

- o 133A dc-to-dc Converter--Converts -48V to +5V. One converter is required.

Cooling Unit

The cooling unit provides cooling, filters incoming air, and circulates air within the cabinet. The cooling unit is located directly below the disk file controller. It is divided into three sections; each containing a fan, control circuitry, filter, and an alarm LED.

D. Power Conditioning Cabinet

The power conditioning cabinet (Fig. 4.9) consists of the following units:

- o Power Distribution Panel

- o Control Panel

- o Rectifier

- o Batteries.

Power Distribution Panel

The power distribution panel contains six fuse blocks. Each fuse position consists of a load fuse and a pilot fuse. Also, this panel is used to distribute the -48V source to the other cabinets.

Control Panel

The control panel provides for the following:

- o Monitoring 208 Vac three phase
- o Monitoring rectifier dc output
- o Charge circuit
- o Charge probe to charge capacitors of the fused lines from the power distribution panel
- o Talk interface between cabinets in the system.

Rectifier

The rectifier rectifies the 208 Vac three phase input voltage for a -48 Vdc (100 amps) output. The rectifier also contains a control panel to monitor the -48V output and a manual switch for ON/OFF functions of the rectifier.

Batteries

Four 6-cell, 12-volt, maintenance-free batteries are provided as a backup/standby for the rectifier in the event the 208 Vac three phase is interrupted. These batteries are connected in series and provide a -48V backup for approximately 15 minutes.

E. Moving Head Disk Transport

The moving head disk transport provides for the rapid storage and retrieval of large amounts of data which cannot be economically stored in the main store unit in the processor cabinet.

A choice of two models of moving head disk transports is provided in the 3B20S Processor. These are:

- o KS-22072, L1 (300 Megabyte)
- o KS-22072, L2 (300 Megabyte).

Moving head disk transports are interfaced to the processor via the disk file controller. These transports may be connected together by a daisy chain since each disk file controller can support a maximum of eight moving head disk transports.

The KS-22072, L1 and KS-22072, L2 disk drives have storage capacities of 300 megabytes. The disk packs used in these drives consist of 12 recording disks. The top and bottom disks are used for protection only. The remaining 10 disks provide 19 data surfaces and one prerecorded servo surface.

F. Video Terminal

The video terminal provides the user interface to the 3B20S Processor which is used for system operations and maintenance. It is interfaced to the processor via an input/output processor unit.

The 4025BS/001/AF video terminal is an uppercase and lowercase (split screen) ASCII terminal having a variety of controllable characters and screen options. These include:

- o Split screen capability
- o Ability to set baud rates
- o Tabs
- o Answer back message from the keyboard.

G. Growth Cabinet

A fully equipped growth cabinet (Fig. 4.10) contains the following units:

- o Input/Output Processor Basic Unit
- o Input/Output Processor Growth Unit
- o 4-inch Cooling Unit
- o Disk File Controller
- o 8-inch Cooling Unit.

H. Growth Units

In addition to adding a growth frame, the capacity of the 3B20S Processor can be expanded by adding the following units:

- o Input/Output Processor Basic Unit (1) and Growth Units (2) (Input/Output Peripheral Control Cabinet)
- o Direct Memory Access Channels (Direct Memory Access Unit) (Maximum of 4)
- o Input/Output Channels (Direct Memory Access Unit) (Maximum of 5)

- o Main Store Memory Arrays (Main Store Unit) (Maximum of 16; 8 Megabytes)
- o Moving Head Disk Transports (Maximum of 8 for each disk file controller installed)
- o Video Terminals
- o Line Printers.

INTERFACES

A. Cabling

The 3B20S Processor has two types of cabling; power and input/output signaling. All signal cabling between cabinets is via port holes in the side of each cabinet. Connections from a cabinet to a moving head disk transport, video terminal, or printer is via an overhead cable duct or, if the system is installed on a raised floor, the cabling may be placed under the floor. Power cabling between the 3B20S Processor cabinets is routed through a single duct on the floor or under a raised floor.

B. Signal Interfaces

Figure 4.11 illustrates a typical signal interface for the system.

Processor Cabinet

The units in the processor cabinet are interconnected via buses. The central control is interfaced to all units in the processor cabinet (direct memory access unit, main store, and associated input/output channels). The main store is interfaced to the central control and the direct memory access unit. The direct memory access unit is interfaced to the central control, main store, and to the controllers located in other system cabinets (input/output processor, disk file controller, and peripheral devices using the dual serial channels of the direct memory access facility). The input/output channels are interfaced to the central control and the peripheral devices in the application requiring the use of these input/output channels. The emergency action interface is interfaced to the maintenance video terminal via a cable to the maintenance teletype controller and by a cable to the video terminal.

Input/Output Peripheral Control Cabinet

The input/output processor basic unit is interfaced to the direct memory access unit (processor cabinet) and to the peripheral devices requiring its services via the peripheral controllers. When used with the input/output processor growth unit, the basic unit is interfaced to the growth unit to provide

control functions for the growth unit. The growth unit is also interfaced to peripheral devices requiring its services via peripheral controllers.

Disk Controller/Magnetic Tape Cabinet

The tape transport is interfaced to the input/output processor via a tape transport peripheral controller. The disk file controller is interfaced to the direct memory access unit in the processor cabinet and to the moving head disk transports under its control. The maximum length of cable between the disk file controller and the last dedicated moving head disk transport is 50 feet.

Power Conditioning Cabinet

The power conditioning cabinet provides the dc power distribution for the system. It is interfaced to the commercial source voltage and to all cabinets in the 3B20S Processor. Buses are provided for internal connections between the units in the cabinet.

Peripheral Units

The basic peripheral units of the 3B20S Processor are the video terminal, printer, tape transport, and moving head disk transport. The video terminal, printer, and tape transport are interfaced to the system via peripheral controllers in an input/output processor unit. Moving head disk transports are interfaced to the system via a disk file controller. Also, the moving head disk transport can be interfaced to additional moving head disk transports via a daisy chain. Two cables are required; a dedicated cable for the disk file controller and a common control (daisy chain) cable. The maximum cable length between the last moving head disk transport and the disk file controller is 50 feet. Additional peripheral devices may be interfaced to the system via peripheral controllers in the input/output processor units.

POWER REQUIREMENTS

The 3B20S Processor requires ac source voltages of 117 Vac single phase, 208 Vac three phase, and 208 Vac single phase. The 117 Vac is provided through the service outlets for units such as the video terminal, printer, test equipment, etc. The 208 Vac three phase is the source voltage for the rectifier which supplies the -48 Vdc for the system. A balanced 208 Vac single phase is provided for the moving head disk transports. The batteries (four 12 Vdc) supply a 15 minute backup for the -48 Vdc. These are switched into operation upon interruption of the 208 Vac three phase power.

The processor cabinet contains a filter for the -48 Vdc input from the power conditioning cabinet. The filter is located at

the bottom of the cabinet and is accessible only from the back.

A. Processor Cabinet

The processor cabinet requires -48 Vdc for the dc-to-dc converters and the cooling unit. The dc-to-dc converters provide the following voltages for the units in the processor cabinet:

- 244D converter: +5V (maximum of four with outputs in parallel)

- J1C129AE power unit: contains one ED-4C188-30 reference converter (+5V, +12V, and -12V) and one 132AJ converter (-5V, -12V, and +12V).

B. Input/Output Peripheral Control Cabinet

The input/output peripheral control cabinet requires -48 Vdc for the cooling units and dc-to-dc converters in the input/output processor units. These dc-to-dc converters are as follows:

- Basic Unit: three 136N converters (+5V)

- Growth Unit: one 136N converter (+5V).

C. Disk Controller/Magnetic Tape Cabinet

The disk controller/magnetic tape cabinet requires -48 Vdc for the tape transport, cooling units, and the 133A (+5V) dc-to-dc converter in the disk file controller.

D. Growth Cabinet

The growth cabinet requires -48 Vdc for the cooling units, the 136N (+5V) converters in the input/output processor units, and the 133A (+5V) converter in the disk file controller.

E. Power Control

A control panel for control of power distribution is provided on the power conditioning cabinet. The power for the processor cabinet, disk file controller, and input/output processor can be controlled by the power switch (ABB1) located in these units. The tape transport, video terminal, printer, and moving head disk transports all have ON/OFF switches.

SPACE/ENVIRONMENT

A. Space

The 3B20S Processor cabinets are 6 feet high and each cabinet requires approximately 4 square feet of floor space. For each moving head disk transport in the system, approximately 6 square

feet is required. The video terminal and printer set on separate pedestals and each requires approximately 2 square feet of floor space.

Additional space is also necessary for accessibility to the system. There should be an aisle (minimum of 2 feet 6 inches) on all sides of the cabinet lineup. A minimum of 4 feet must be available in front of a moving head disk transport for removal of the transport from the lineup.

B. Environment

The 3B20S Processor is designed to be functional within the following ranges:

- o Temperature: +40F to +100F (Normal operating temperature: +75F to +78F)
- o Relative Humidity: 20 to 55%
- o Heat Dissipation: 80 watts per square foot
- o Floor Load: Not to exceed 140 pounds per square foot.

Systems installed at locations designated as an earthquake area are provided with earthquake mounting apparatus.

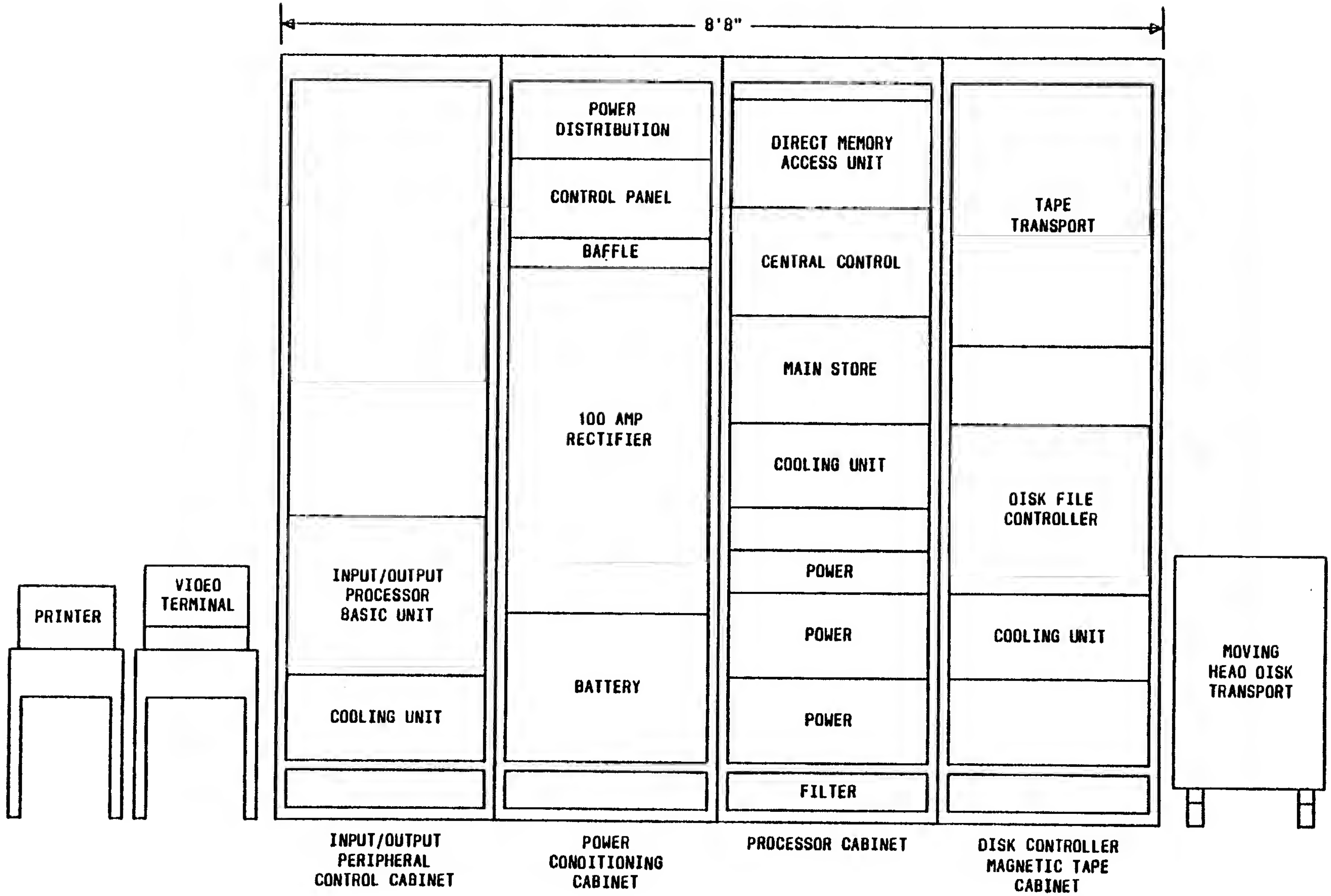

Fig. 4.1--3B20S Processor Minimum Configuration

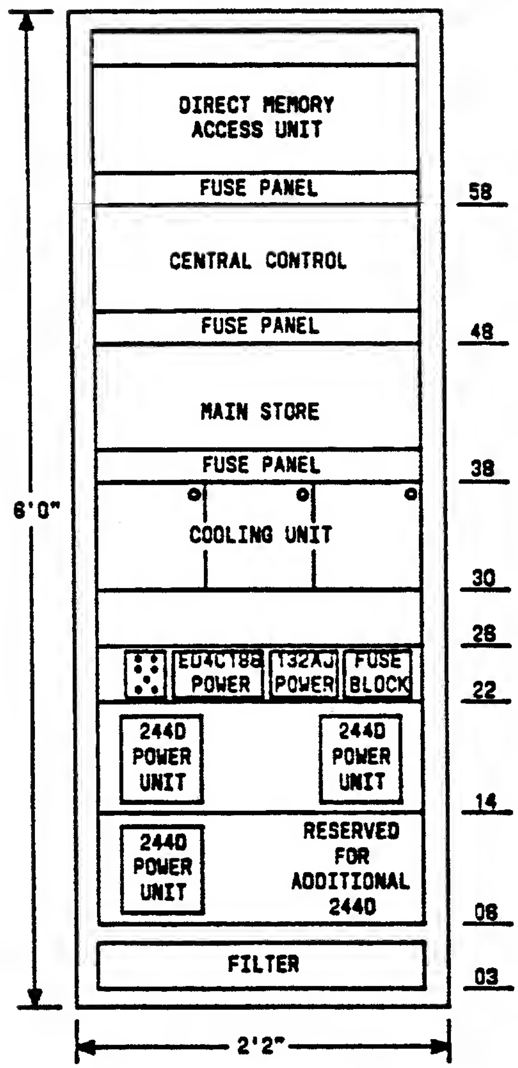

Fig. 4.2--Processor Cabinet

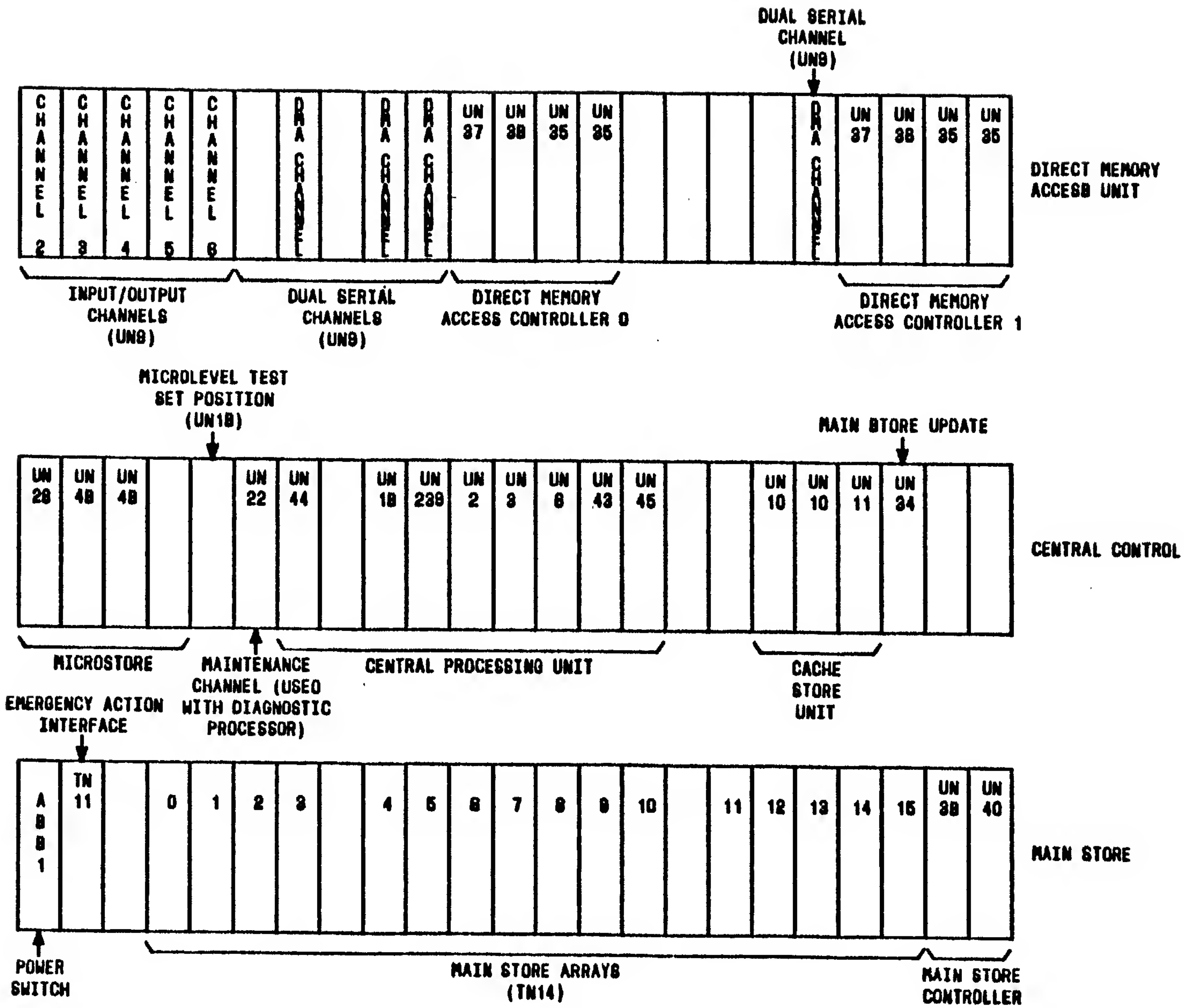

Fig. 4.3--Processor Cabinet (Circuit Pack Layout)

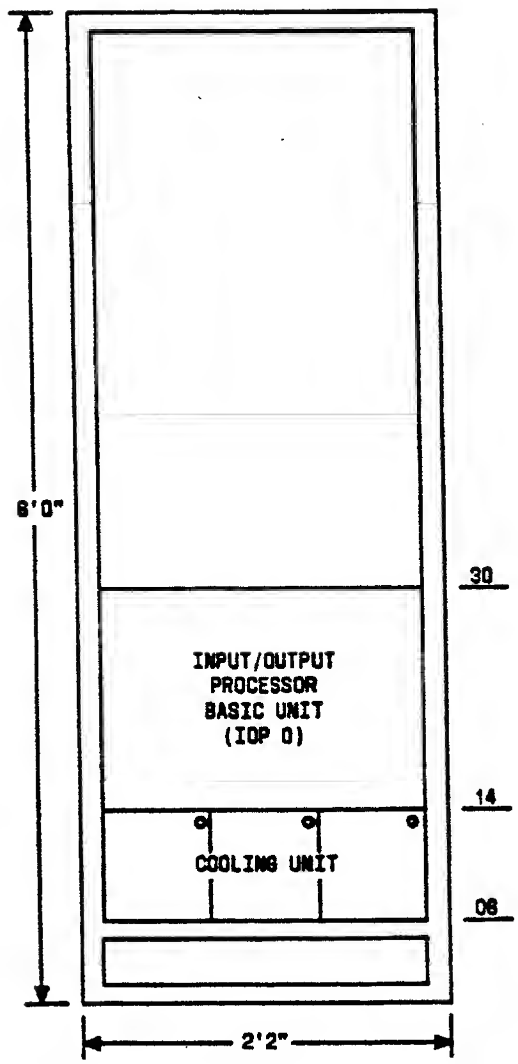

Fig. 4.4--Input/Output Peripheral Control Cabinet (Minimum Configuration)

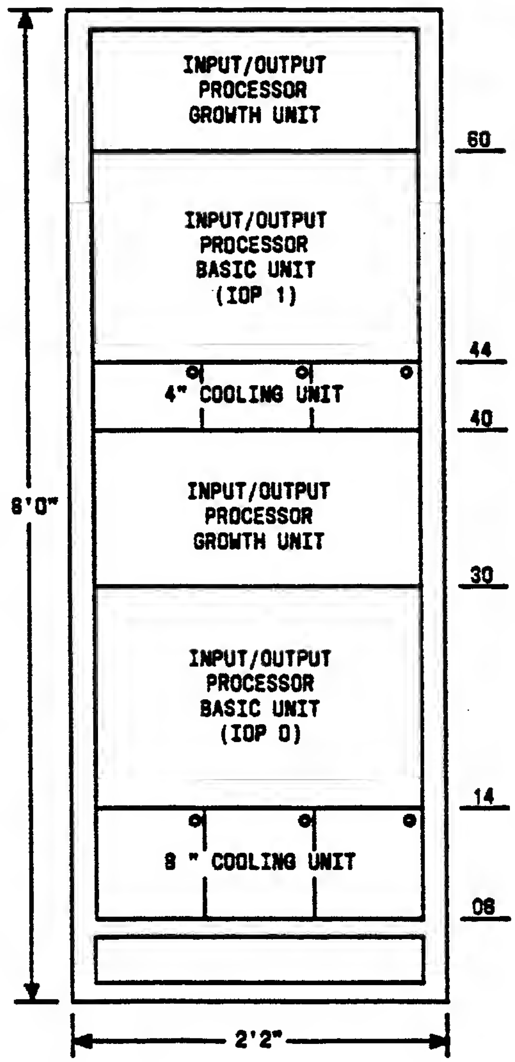

Fig. 4.5--Input/Output Peripheral Control Cabinet (Fully Equipped)

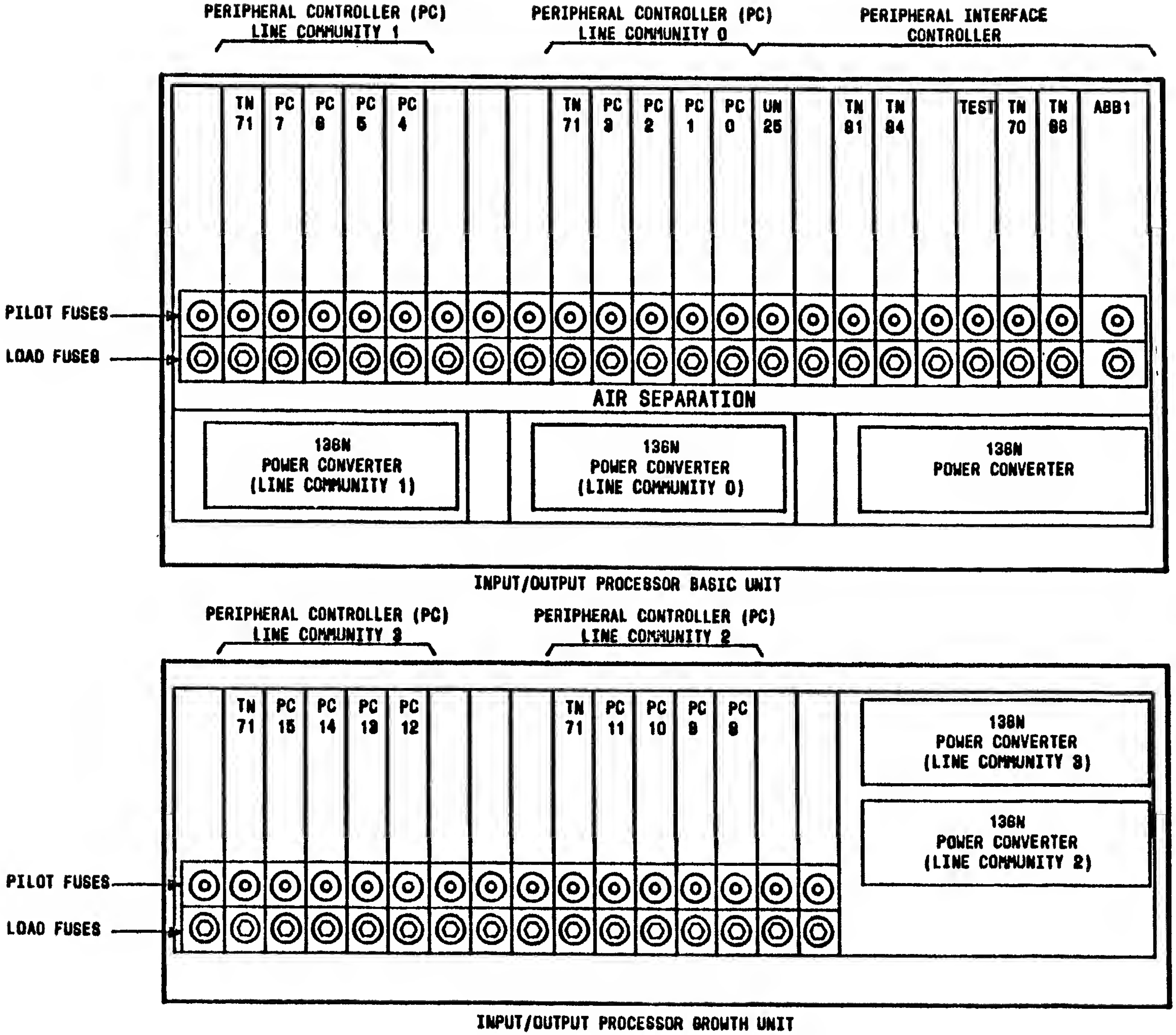

Fig. 4.6--Input/Output Processor Basic and Growth Units (Circuit Pack Layout)

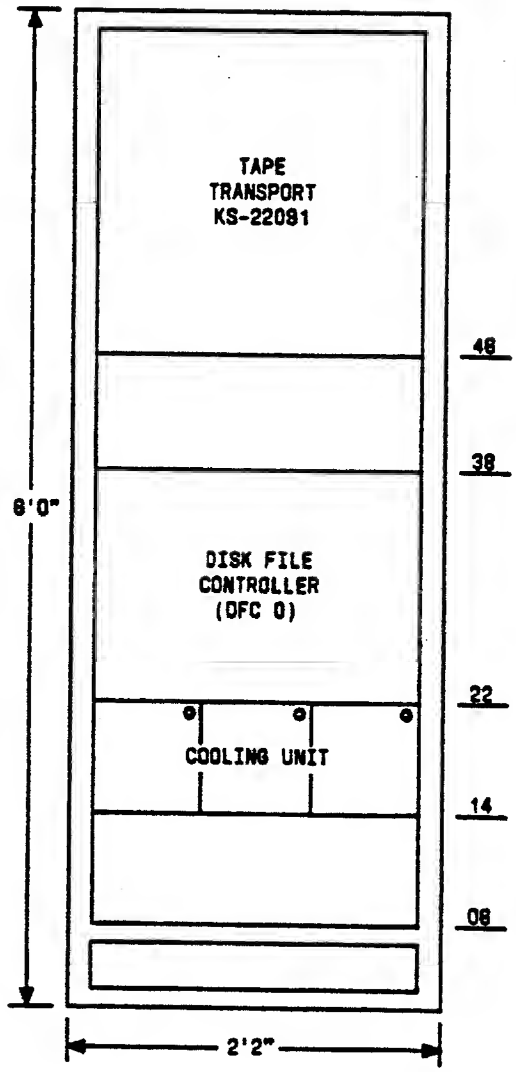

Fig. 4.7--Disk Controller/Magnetic Tape Cabinet

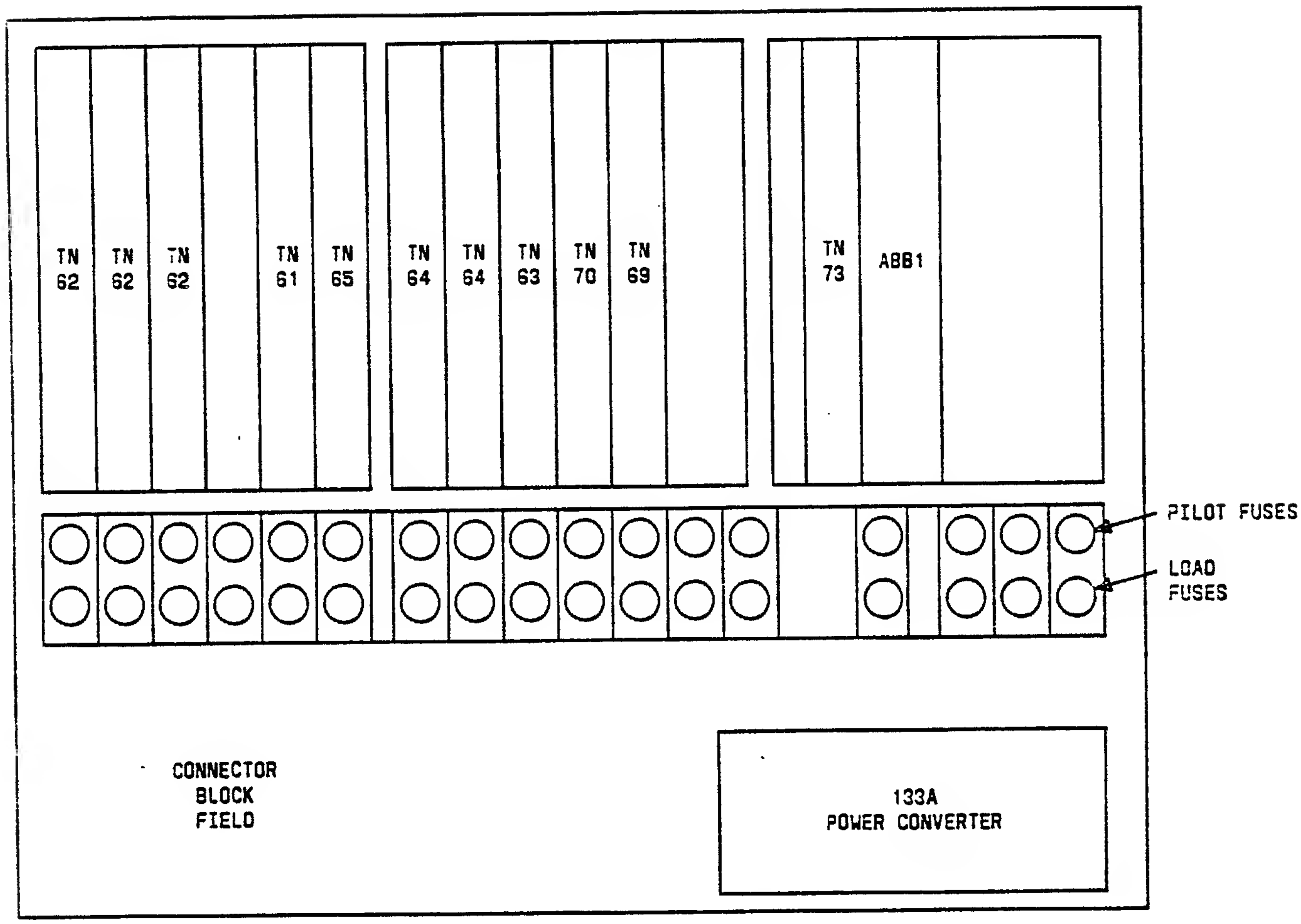

Fig. 4.8—Disk File Controller (Circuit Pack Layout)

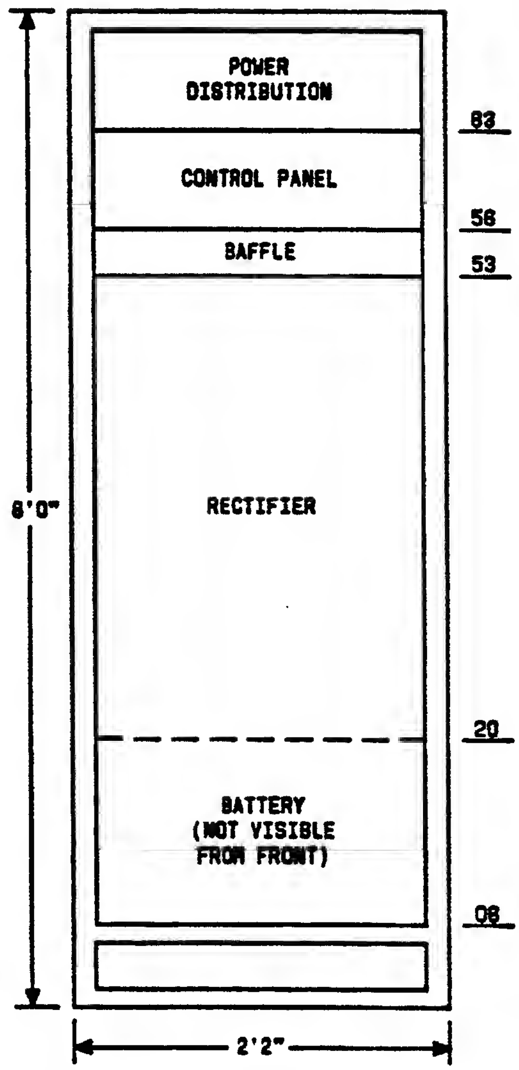

Fig. 4.9--Power Conditioning Cabinet

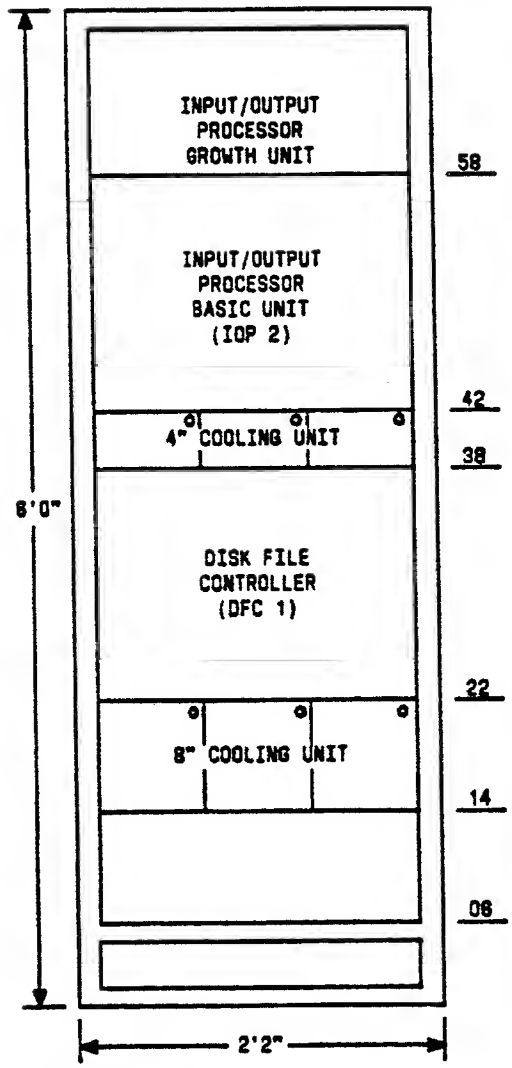

Fig. 4.10--Growth Cabinet (Fully Equipped)

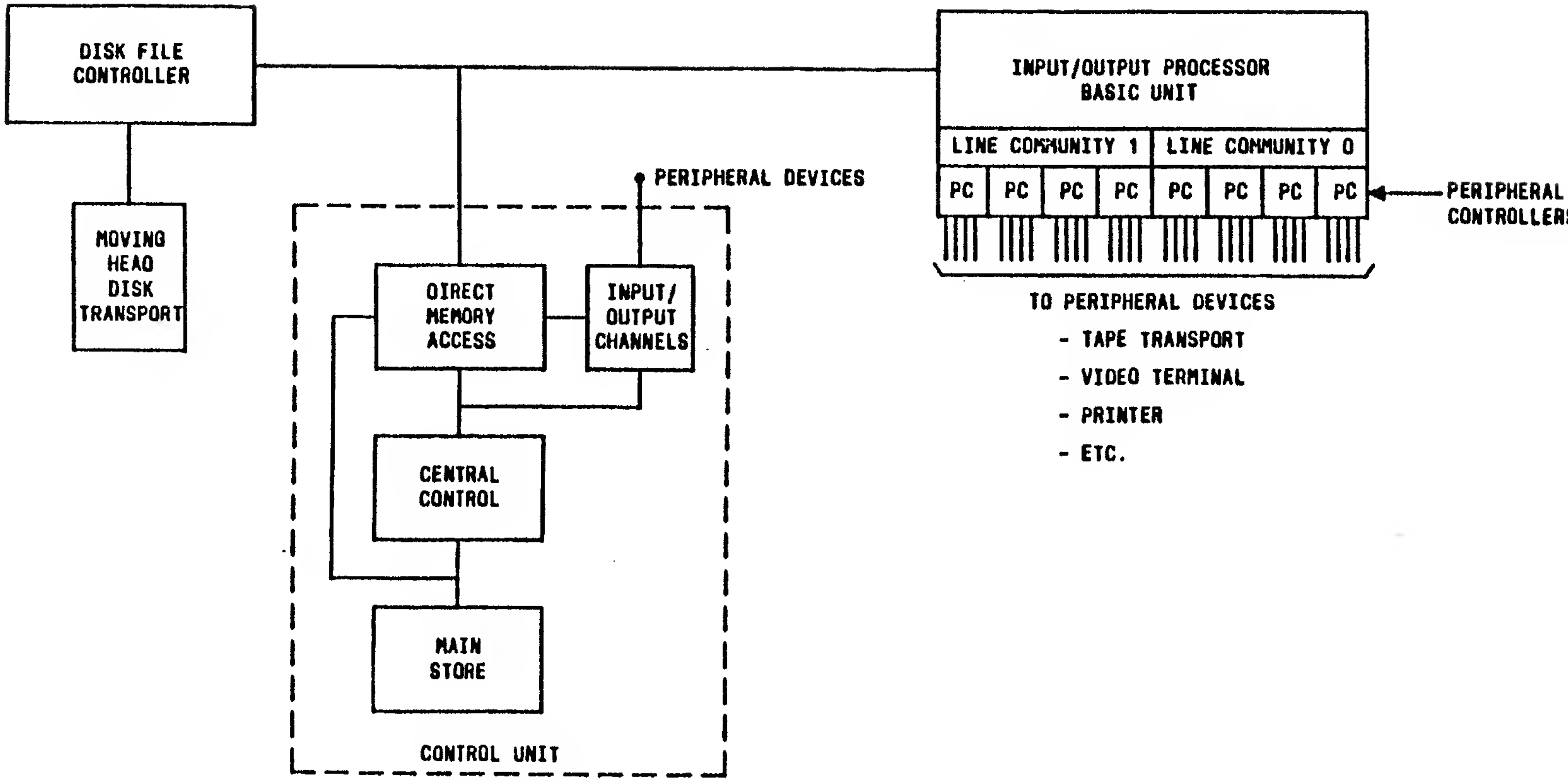

Fig. 4.11--3B20S Processor Signal Interfaces

TABLE A - 3B20S PROCESSOR DIMENSIONS

EQUIPMENT	UNIT	WIDTH	HEIGHT
PROCESSOR CABINET 2'2" X 2' X 6'	DIRECT MEMORY ACCESS CENTRAL CONTROL MAIN STORE COOLING UNIT	26" 26" 26" 26"	8" 8" 6" 6"
INPUT/OUTPUT PERIPHERAL CONTROL FRAME 2'2" X 2' X 6'	INPUT/OUTPUT PROCESSOR BASIC UNIT* INPUT/OUTPUT PROCESSOR GROWTH UNIT* COOLING UNIT*	26" 26" 26"	18" 10" 6"
POWER CONDITIONING CABINET 2'2" X 2' X 6'	POWER DISTRIBUTION PANEL CONTROL PANEL RECTIFIER BATTERY	26" 26" 26" 26"	7" 7" 33" 12"
DISK CONTROLLER MAGNETIC TAPE CABINET 2'2" X 2' X 8'	TAPE TRANSPORT DISK FILE CONTROLLER* COOLING UNIT	26" 26" 26"	24" 16" 6"
PERIPHERAL UNIT	MOVING HEAD DISK TRANSPORT VIDEO TERMINAL	26" 15.5"	40" 14.5"

*USED IN GROWTH CABINET

NOTES

5. FUNCTIONAL DESCRIPTION

SECTION PURPOSE

This section of the system description describes the functions of individual 3B20S Processor units. Descriptions of internal unit functions as well as interactions with other units are included.

The information is divided into the following five equipment areas:

- o Processor Units
- o Peripheral Control Units
- o Power
- o Peripheral Units
- o Cooling Unit.

PROCESSOR UNITS

The processor units provide the control functions for the 3B20S Processor. The processor units are:

- o Direct Memory Access Unit
- o Central Control
- o Main Store
- o Power Unit.

A. Direct Memory Access Unit

The direct memory access unit consists of the following subunits:

- o Direct Memory Access Controller
- o Direct Memory Access Channels
- o Input/Output Channels.

Direct Memory Access Controller

The direct memory access controller (Fig. 5.1) interfaces to the central control and main store via internal processor buses. It is interfaced to the input/output processor units, disk file

controller, and other designated peripheral units via the direct memory access controller channels (dual serial channels) via intercabinet signaling cables. A dual serial channel can interface to a maximum of 16 peripheral devices.

The direct memory access controller enables the transferring of blocks of data directly between the main store and the peripheral units (via the direct memory access channels) without the central control performing each individual word transfer. Communications to and from the direct memory access controller is in parallel form and active low (0 volts).

Direct Memory Access Channels

The direct memory access channels are equipped with dual serial channels (Fig. 5.2). The dual serial channels provide the interface between the direct memory access controller and peripheral devices using the direct memory access functions. The dual serial channel is a semiautonomous unit providing an interface to a maximum of 16 peripheral devices. Differential dc line driving and receiving circuits are used to provide a serial data path for each peripheral device.

Each peripheral device interfaces the dual serial channel via five signal paths. These are:

- Two bidirectional data paths
- One transmit clock path
- One receive clock path
- One request path.

Data message transmission is preceded by the start code. All start codes begin with a leading logic 1 (5 volts) followed by a 1-out-of-3 code. The start code specifies the peripheral device operation. The return code transmitted by the peripheral device to the dual serial channel specifies the success or failure of the requested operation.

The dual serial channel provides for the transmission of address, control, and data transfers between the direct memory access controller and the peripheral devices using the direct memory access. It also receives status and reply data from the peripheral device and transmits these to the direct memory access controller.

Input/Output Channels

The 3B20S Processor can have a maximum of five input/output channels. These channels are dual serial channels and are located in the direct memory access unit.

The input/output channels interface peripheral devices to the

central control unit. Each input/output channel is operated under the central control software. Address, control, and data information is transmitted from the central control to the input/output channels and the designated peripheral devices.

The input/output channels are designed to function with low-, medium-, and high-speed peripheral devices. The dual serial channel is a semiautonomous input/output channel providing an interface between the processor and a maximum of 16 high-speed peripheral devices.

B. Central Control

The central control (Fig. 5.3) interfaces to the main store, input/output channels, and the direct memory access unit via internal processor buses. The central control contains the following subunits:

- o Microstore
- o Central Processing Unit
- o Main Store Update Unit
- o Cache Store Unit.

Microstore

The central control is a microprogrammed processor with a read-only memory that contains a series of microinstructions which direct the activity of the 3B20S Processor. Each main store instruction is put into effect by executing a series of microinstructions that are read out of microstore, one microinstruction at a time.

The microstore is addressed using a 16-bit microaddress and normally controlled by the microcontrol section of the central control. Output of the microstore is a microinstruction (56 bit plus 8-parity bit) word which is decoded by microcontrol to provide control signals for the central control.

Fields within the microinstruction are used to define the type of microinstruction, minimum microinstruction execution time (150, 200, 250, or 300 nanoseconds), source and destination registers, operations to be performed, the address of the next microinstruction, and check bits to verify operation of the microstore.

Central Processing Unit

Sixteen 32-bit general registers are available and are usable by the software as a scratch pad as well as by the central control as source and destination registers for data. The arithmetic logic unit in the data manipulation unit performs arithmetic and logical operations. These operations are binary

2s complement addition and subtraction, OR, AND, EXCLUSIVE OR, EXCLUSIVE NOR, and complement. The rotate mask unit in the data manipulation unit provides right rotates and shifts of 0 to 31 bits in one operation. The AND/OR operations can be performed on bits, nibbles (4-bit groups), bytes (8 bits), and half words (16 bits).

Main Store Update Unit

The main store update unit controls access to the main store. Since the main store can not be accessed by the central processing unit and the direct memory access unit at the same time, the main store update unit receives the main store access requests and controls the order in which the requesting units receive access to the main store.

Cache Store Unit

The cache store unit provides a means of decreasing the access time of main store operations. The unit consists of three high-speed memory arrays, each containing 512 words, and has an access time of 250 nanoseconds. Each word read from the main store is also stored in the cache store unit. Each time a main store read is requested part of the address is decoded to read the corresponding address in the cache memory. If the requested data word is in the cache memory, it is returned from there and the main store read is inhibited.

C. Main Store

The main store (Fig. 5.4) is interfaced to the central control and the direct memory access unit via internal processor buses. The main store is comprised of the main store controller and a maximum of 16 main store arrays. The main store controller interfaces units connected to the main store arrays. The main store array is the storage area for program instructions and data. Each array can store 128K 40-bit words. The minimum 3B20S Processor configuration contains two main store arrays.

Control of the main store is provided by the central control or the direct memory access unit. Operations performed by the main store are:

- o Read (full word)
- o Read and clear (full word)
- o Read and clear (half word)
- o Read and clear (byte)
- o Write (full word)
- o Write (half word)

- o Write (byte)
- o Memory refresh
- o Memory error detection.

Two additional circuit packs are located in the main store. These are the power switch (ABB1) and the emergency action interface (TN11). The power switch provides manual control of the power in the processor cabinet. The emergency action interface provides a maintenance access to the 3B20S Processor via a video terminal. It also monitors the dc-to-dc converters in the processor cabinet.

D. Power Unit

The power unit in the processor cabinet is comprised of the following units:

- o A maximum of four 244D dc-to-dc converters (-48V to +5V). The +5V outputs are paralleled to increase current capability.
- o J1C129AE power unit--This unit consists of an ED-4C188-30 dc-to-dc converter (-48V to +5V, +12V, and -12V) and a 132AJ dc-to-dc converter (-48V to -5 and +12V).

The 244D dc-to-dc converter has an ON/OFF switch, a low voltage alarm LED, and a monitoring circuit for input/output voltages and output current. The +5V output is used by the logic circuits in the processor.

The 132AJ converter supplies memory voltages of -5V and +12V for the main store. The ED-4C188-30 supplies -5V, +12V, and -12V for the emergency action interface circuit.

PERIPHERAL CONTROL UNITS

The 3B20S Processor has three basic peripheral control units:

- o Input/Output Processor Basic Unit
- o Input/Output Processor Growth Unit
- o Disk File Controller.

A. Input/Output Processor Basic Unit

The input/output processor basic unit (Fig. 5.5) interfaces with the 3B20S Processor via the direct memory access unit. It also interfaces to peripheral devices via peripheral controllers in its line communities. When used in conjunction with an input/output processor growth unit, it is interfaced to the growth unit to provide control functions.

The input/output processor consists of three groups of subunits:

- o Peripheral Interface Controller
- o Peripheral Line Community 0
- o Peripheral Line Community 1.

The peripheral interface controller provides the interface to peripheral controllers and the direct memory access unit. It consists of the following circuits:

- o Power Switch (ABB1)
- o Duplex Dual Serial Bus Selector
- o Bus Interface Controller
- o Peripheral Interface Controller
- o Peripheral Interface Microcontrol Store
- o Input/Output Microprocessor Interface
- o 136N dc-to-dc Converter.

Each peripheral line community consists of the following units:

- o Up to four Peripheral Controllers
- o Power Monitor
- o 136N dc-to-dc Converter.

The input/output processor basic unit functions as a front-end processor to control input/output transfers between the 3B20S Processor and various peripheral devices, thereby reducing the load on the processor. A 16-bit bipolar microprocessor (peripheral interface controller) interfaces the processor with up to four peripheral line communities. Each peripheral line community can contain four individual peripheral controllers, and each peripheral controller may be equipped with a multiple of peripheral devices (depending upon the types of peripheral controllers and devices). The peripheral interface controller will autonomously transfer blocks of data between peripheral controllers and the processor. Also, operation controls and status are transmitted between the peripheral controllers and the processor.

The input/output processor basic unit contains a 136N dc-to-dc converter (-48V to +5V) for its power requirements. Also, it contains a power switch (ABB1) for the manual control of power.

This power control is also provided for the input/output processor growth unit when interfaced to a basic unit.

B. Input/Output Processor G-owth Unit

The input/output processor growth unit must be used with a basic unit to be functional. The basic unit provides the control functions (data and power) for the growth unit. The input/output growth unit expands the capability of the basic unit by the addition of two peripheral line communities, providing the additional capacity for eight peripheral controllers.

The input/output growth unit consists of two peripheral line communities. Each community consists of the following:

- o Up to four Peripheral Controllers
- o Power Monitor
- o 136N dc-to-dc Converter.

C. Disk File Controller

The disk file controller (Fig. 5.6) interfaces with the 3B20S Processor (via the direct memory access unit) and a maximum of eight moving head disk transports.

The disk file controller consists of the following subunits:

- o Dual Duplex Serial Bus Selector
- o Bus Interface Controller
- o Peripheral Interface Controller
- o Peripheral Interface Microcontrol Store
- o Parallel Serial Data Interface
- o Moving Head Disk Control
- o Moving Head Disk Data Clock
- o Power Monitor/Converter
- o Power Switch (ABB1)
- o 133A dc-to-dc Converter (-48V to +5V).

The disk file controller facilitates 2-way communication between the central control (via the direct memory access unit) and the moving head disk transports. The central control accesses the moving head disk transport via the disk file controller to restore main store memory, store data, and access stored data as required.

The dual duplex serial bus selector interfaces the disk file controller to the direct memory access unit. The bus interface controller is used to interface the dual duplex bus selector to the peripheral interface controller. The peripheral interface controller is a microprocessor that controls data flow between the moving head disk transport and the direct memory access unit. The peripheral interface microcontrol store is part of the peripheral interface controller circuitry and provides decoding of command signals and formats the requested orders for use by the peripheral interface controller. The parallel serial data interface converts parallel data to serial data for communication to the moving head disk transport and converts serial data to parallel data for communications from the moving head disk transport to the peripheral interface controller. The moving head disk control interfaces the disk file controller to the moving head disk transport and supplies the data to select the transport, head position, read/write mode, and monitor transport status. The moving head disk data clock interfaces the parallel serial data interface with a maximum of eight moving head disk transports. It also ensures that only one transport is selected at a time and formats data for the transport.

The power monitor/converter (TN73) monitors output voltage from the 133A dc-to-dc converter and converts -48V to -5V for the moving head disk control and moving head disk data clock. The 133A dc-to-dc converter (-48V to +5V) supplies the voltage required for the disk file controller. The power switch (ABB1) provides manual control for the disk file controller.

POWER

The units in the 3B20S Processor that are provided with a dc-to-dc converter(s) are:

- o Processor Units
- o Input/Output Processor Basic Unit
- o Input/Output Processor Growth Unit
- o Disk File Controller.

These units are provided manual power control via a power switch (ABB1).

The tape transport operates from -48 Vdc, the moving head disk transport operates from 208 Vac (single phase), and the video terminal and printer operate using 117 Vac. Each of these units are provided with a manual ON/OFF power switch.

The power conditioning cabinet operates from commercial 3-phase 208 Vac. It provides -48 Vdc to the 3B20S Processor System. The ac operational voltage is distributed via standard ac

outlets installed at the site.

The power conditioning cabinet (Fig. 5.7) contains the following units:

- o Six fuse blocks
- o Control panel
- o Rectifier
- o Batteries.

The commercial 3-phase 208 Vac is supplied to the rectifier, which rectifies the ac input to -48 Vdc at 100 amperes. The dc output from the rectifier is supplied to the 3B20S Processor cabinets via the six fuse blocks.

The six fuse blocks provide fusing (load and pilot) and distribution of the -48 Vdc from the rectifier to the 3B20S Processor cabinets.

The control panel provides:

- o Monitoring of the 3-phase 208 Vac
- o Monitoring of rectifier voltage
- o Monitoring of the batteries
- o Charging circuit and charge probe for charging capacitive loads on the fuses (fuse blocks)
- o A communication interface between the power conditioning cabinet and other 3B20S Processor cabinets.

The battery section of the power conditioning cabinet is comprised of four 6-cell, 12-volt batteries connected in series to provide -48 Vdc. These batteries are maintenance free and provide a 15-minute backup for the rectifier if commercial ac is interrupted. The battery output will be switched into service automatically and will provide a graceful power down in case of a power interruption to the processor.

The processor cabinet has a filter unit for the dc source voltage. This filter is located below the fuse blocks in the cabinet and is accessible from the rear.

PERIPHERAL UNITS

The primary peripheral units provided with the 3B20S Processor are:

- o 4025BS/001/AF Video Terminal

- o 40P2F Printer
- o Moving Head Disk Transport
- o KS-22091 Tape Transport.

A. Video Terminal and Printer

The 4025BS/001/AF video terminal provides the user with an operational and maintenance interface to the 3B20S Processor. The 40P2F printer provides for a hard copy of all video terminal inputs and outputs. The terminal and printer are interfaced to the system via a peripheral controller located in an input/output processor unit. The video terminal and printer both require 117 Vac 60Hz for source voltage.

The video terminal is an uppercase and lowercase ASCII terminal (keyboard with cathode-ray tube). It has the following characteristics:

- o Split screen capability
- o Ability to set baud rates, tabs, and answer back messages and to store this information
- o Simple graphic characters
- o Reverse video
- o Self-testing capability.

Additional information is provided in the vendor manual shipped with the terminal.

B. Moving Head Disk Transport

The moving head disk transports are interfaced to the 3B20S Processor via the disk file controller. Also, up to eight moving head disk transports can be connected in a daisy-chain arrangement since the disk file controller can support a maximum of eight moving head disk transports. The moving head disk transport requires 208 Vac for source voltage and is equipped with a control panel to control power and operation.

The moving head disk transport provides a storage area for system data. This allows for the use of a smaller main store area. Additional information on the moving head disk transport is provided in the vendor manual shipped with the unit.

The 300-megabyte moving head disk drive contains 20 recording surfaces (10 disks). Nineteen of the twenty surfaces are used for writing and reading, with the remaining prerecorded surface dedicated to head positioning and timing. The read-write head for a particular disk surface may be positioned to 815 positions,

starting from the outer edge and moving toward the center. Each position is called a track and a group of corresponding tracks on ten disks is called a cylinder. Each track is divided into segments called sectors. A sector is the smallest quantity of data written or read by a disk file controller. `All the read-write heads of a moving head disk drive are mechanically ganged. Only one track of a given disk cylinder (data) is written or read at a given time.

C. Tape Transport

The KS-22091 tape transport interfaces with the 3B20S Processor via a peripheral controller in an input/output processor unit. The tape transport provides a convenient means to store (write) and retrieve (read) selected data.

The tape transport accepts a single reel of half-inch tape on a reel up to 10-1/2 inches in diameter. Data is read or written at a speed of 25 inches per second, while a fast forward or reverse speed of 100 inches per second is available. During all write operations, the mode for detecting any recording errors is read-after-write. The tape unit accepts commands for rewind, forward/reverse, read/write, erase, write end-of-file mark, and off-line. Status indications for beginning of tape, end of tape, on-line, write-enable ring, rewinding, and ready are returned to the system.

The tape transport reads or writes nine tracks on standard magnetic tape. Data is recorded at 1600 bits per inch in a phase-encoded format. Each record is written with vertical parity, and the tape controller is capable of correcting single-track errors on reads.

The tape transport requires -48 Vdc for operation. It is provided with a power ON/OFF switch, a control panel, and a test panel. Additional information is provided in the manual shipped with this unit.

COOLING UNIT

Each cooling unit is divided into three cooling sections. Each contains a fan, control circuitry, and an alarm condition LED. The 8-inch cooling units also contain an air filter in the bottom of the unit. The cooling unit requires -48 Vdc for operation. It is implemented in the system so that if the cooling unit becomes inoperative the unit using it will produce an alarm. The cooling unit provides filtering (8-inch only), circulation of incoming air, and cooling for the units in the cabinets.

Cooling units are required for the following units:

o Each Input/Output Processor Basic Unit (cooling unit located directly below unit)

- Each Disk File Controller (cooling unit located directly below unit)

- Processor Units (cooling unit located directly below main store unit).

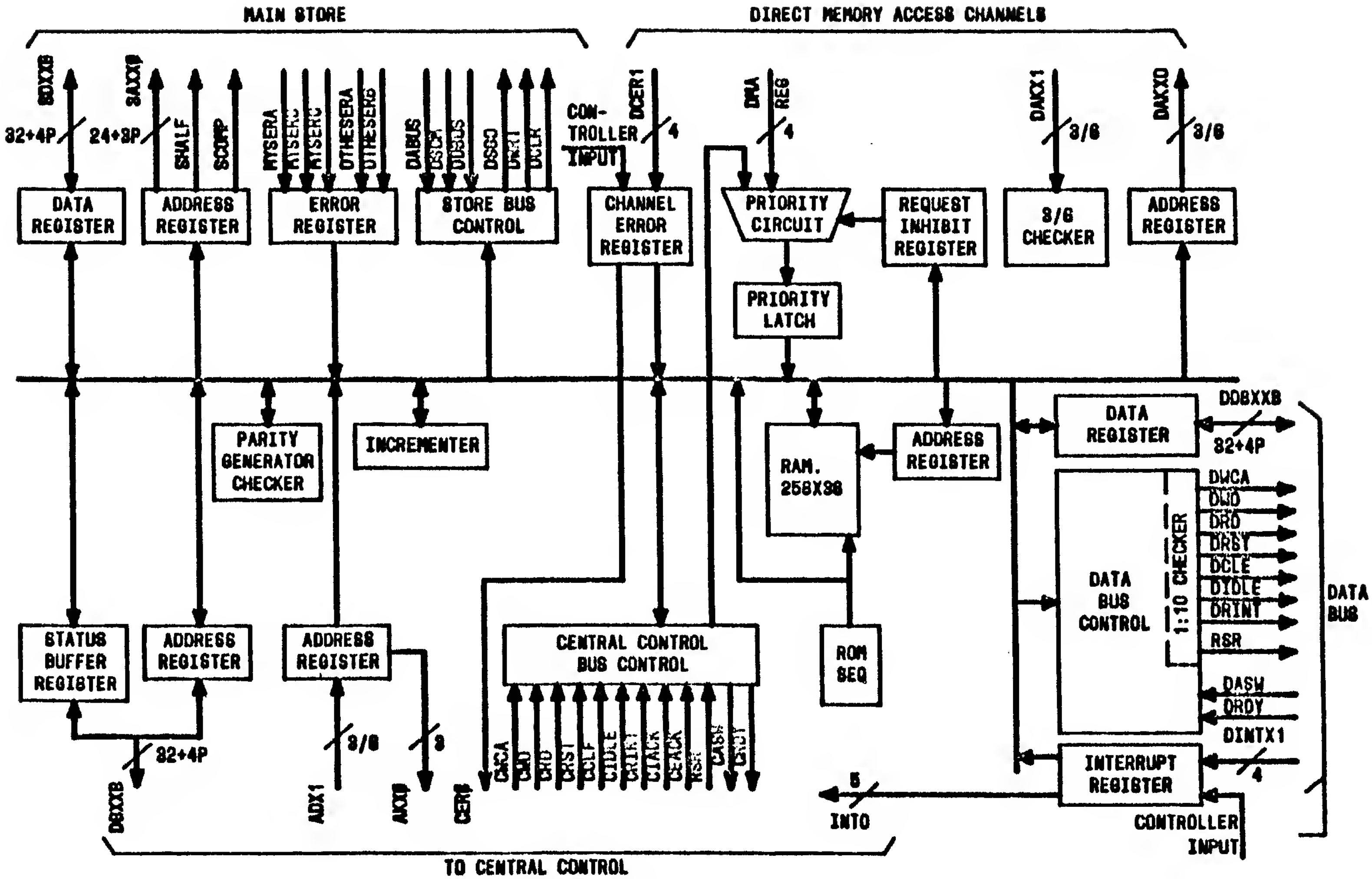

Fig. 5.1--Direct Memory Access Controller - Functional Block Diagram

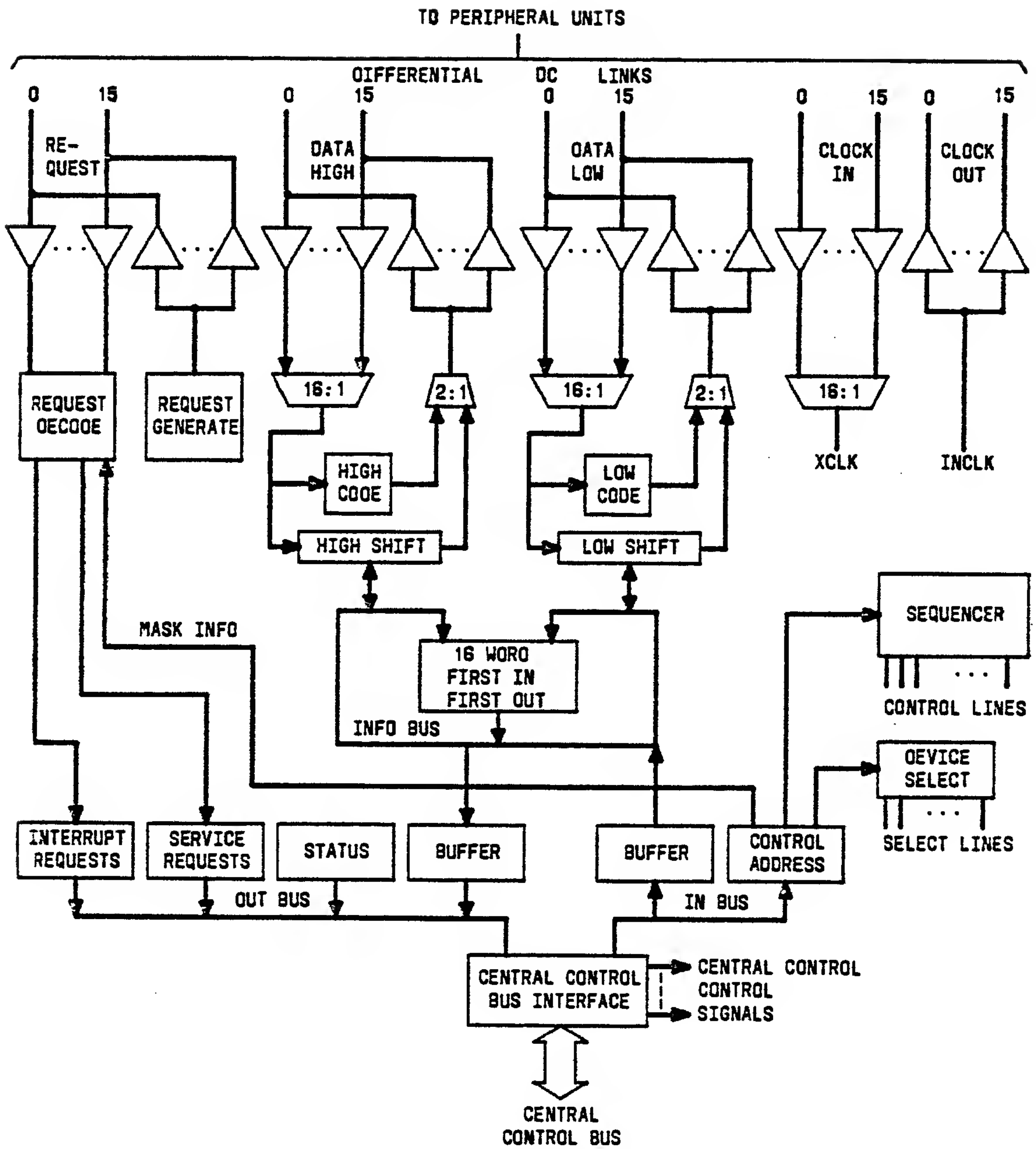

Fig. 5.2--Dual Serial Channel - Functional Block Diagram

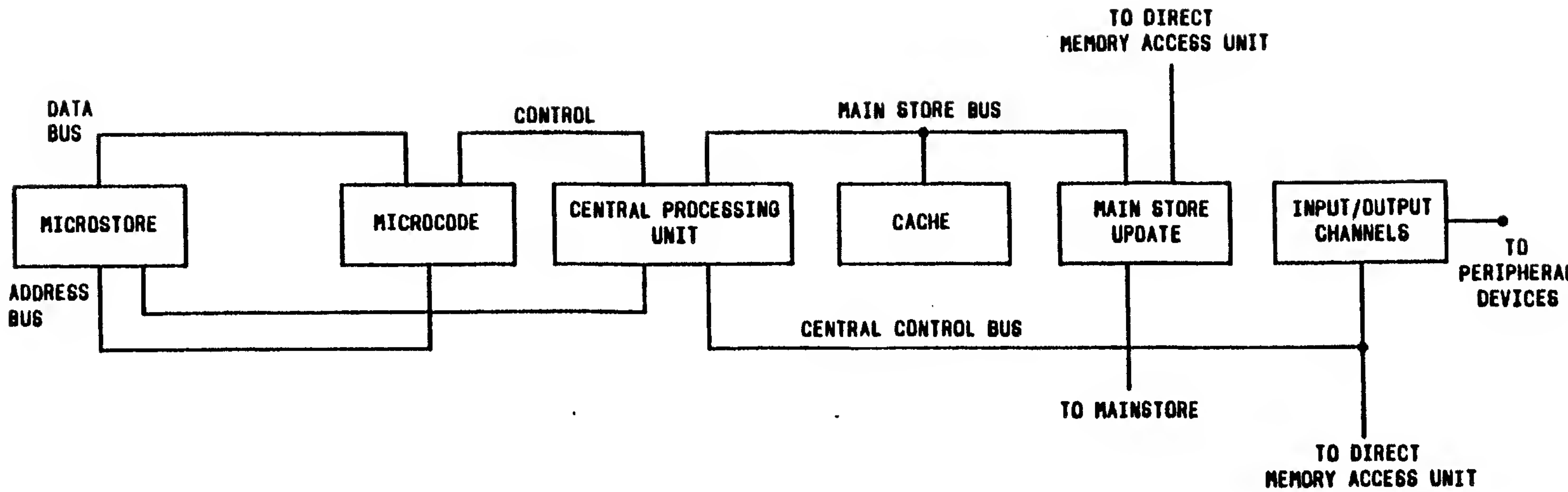

Fig. 5.3--Central Control - Functional Block Diagram

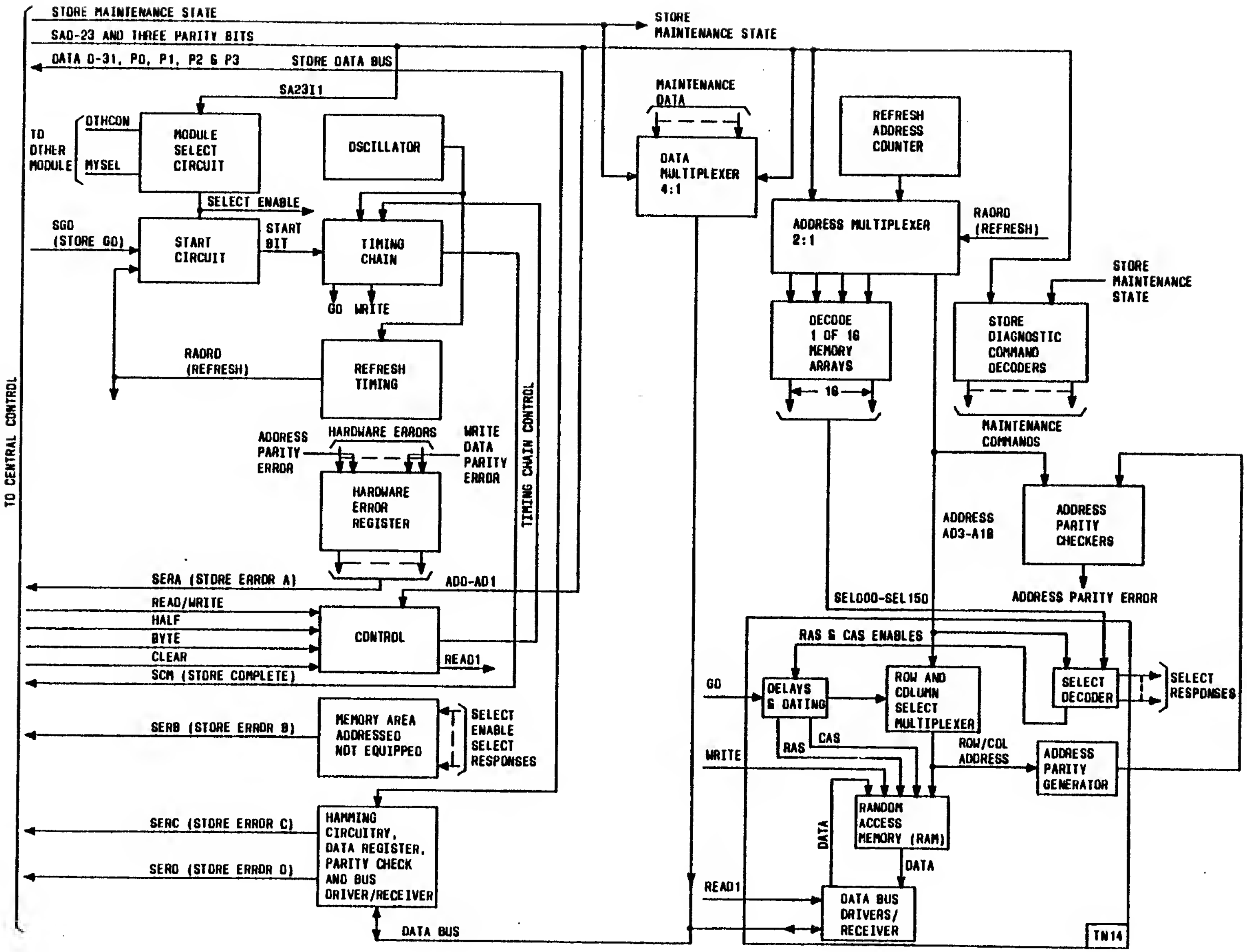

Fig. 5.4--Main Store - Functional Block Diagram

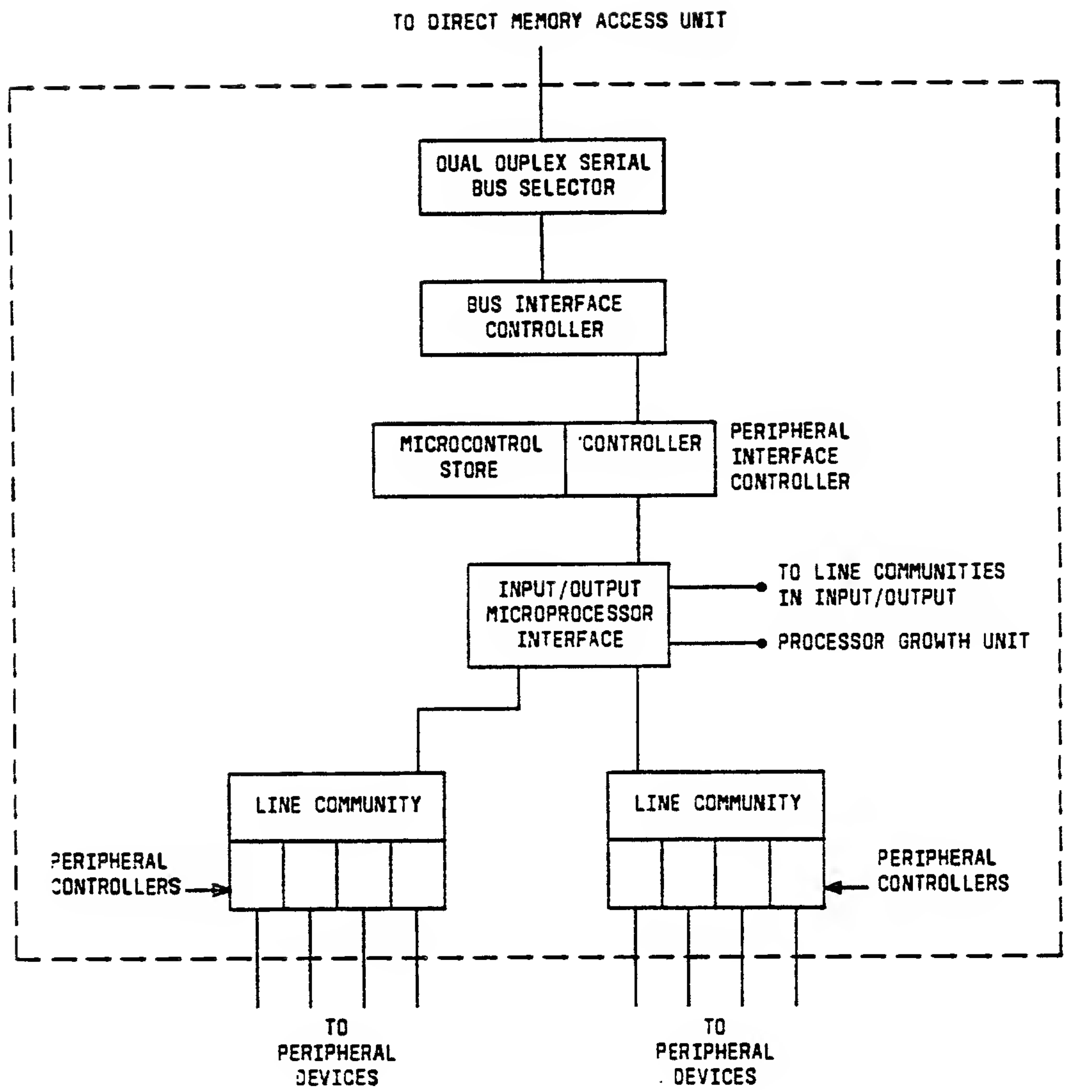

Fig. 5.5--Input/Output Processor Basic Unit - Functional Block Diagram

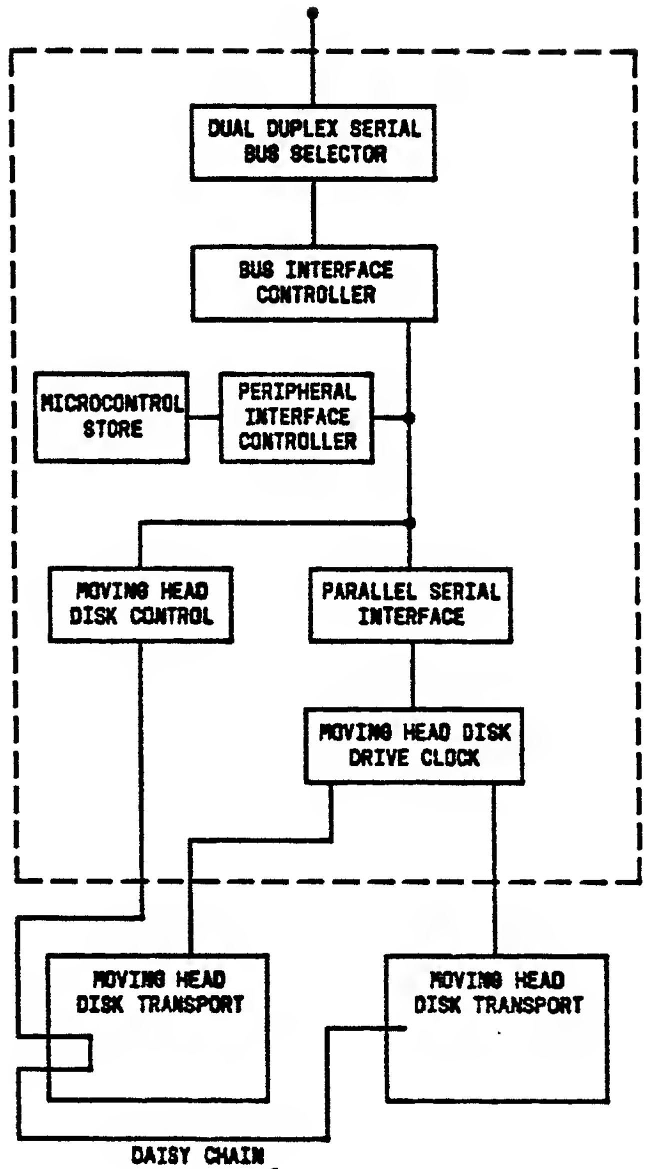

Fig. 5.6--Disk File Controller - Functional Block Diagram

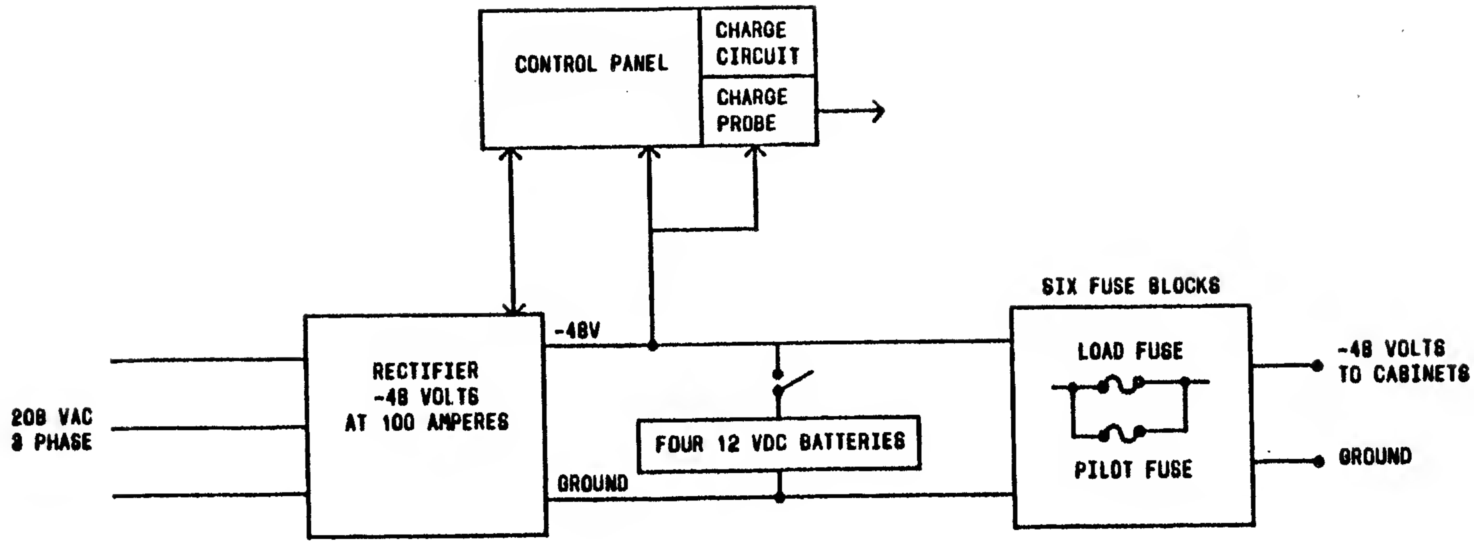

Fig. 5.7--Power Conditioning Cabinet - Functional Block Diagram

NOTES

6. FEATURE DESCRIPTION

SECTION PURPOSE

This section of the system description provides a description of the features which enhance the operation of the 3B20S Processor System.

INTRODUCTION

The 3B20S Processor offers a variety of feature capabilities which can be adapted to the needs of each individual user's application. This versatility, combined with high reliability, makes the 3B20S Processor a very desirable system.

To enhance the reliability of the 3B20S Processor, Western Electric provides support services for the customer selected offering. These offerings can range from full Western Electric support to complete customer independence (customer provides their own equipment service).

Western Electric support services may pertain to software in addition to hardware, thus eliminating the 3B20S Processor customer's need to be concerned with multiple support organizations. This marketing approach provides homogeneity to the 3B20S Processor, allowing features, such as integrated diagnostics, which are not possible with a multisource product, to be made available to 3B20S Processor customers. This approach maximizes operational and support service effectiveness.

Documentation for the 3B20S Processor System includes a set of documents oriented toward the specific user. These documents, delivered with the product, consist of manuals for operating the processor, peripherals, and the operating system. Additional supplementary documentation providing detailed system architectural design and instructions for support operations can be purchased.

SYSTEM FEATURES

The system features of the 3B20S Processor are grouped by association into the following categories:

- o Control Unit
- o Peripherals
- o Diagnostics
- o Power

o System Configuration

o Software/Firmware.

A. Control Unit Features

Processor

The central processing unit of the 3B20S Processor is a self-checking 32-bit processor. Self-checking logic of the processor ensures that errors are detected when they occur, thereby eliminating drastic recovery measures required later in processing if errors go undetected.

The 3B20S Processor uses a 24-bit virtual address, i.e., the virtual address is a main store addressing as seen by the programmer. The main store is addressed with a 24-bit physical byte address. The 24-bit virtual address is converted into a 24-bit physical address using a paged segmentation technique. The paged segmentation technique involves a virtual-to-physical address translation accomplished on a per-page basis using dedicated hardware. The address space is divided into 64 segments of 128K bytes each (K=1024). Each of the 64 segments is divided into 64 pages with each page containing 2K bytes. Protection is on a page level allowing pages within a segment to have different access rights.

The 3B20S Processor is equipped with a maximum of five programmed input/output channels and a maximum of four direct memory access channels. Programmed input/output channels are directed by central control instructions over the central control input/output bus. The direct memory access facility provides the capability for direct memory transfers between the main store and peripheral devices without the aid of the central control. This process reduces the real-time required for the processing of input/output requests. The direct memory access unit consists of one or two direct memory access controllers, each capable of autonomous control of as many as four direct memory access channels through the direct memory access input/output bus.

The basic 3B20S Processor configuration consists of one direct memory access unit equipped with one direct memory access channel. Two additional direct memory access channels can be added in growth situations to the direct memory access controller. If an additional (fourth) direct memory access channel is required, another direct memory access controller can be added to this configuration. This controller can only contain one direct memory access channel. No programmed input/output channels are available in the basic configuration; however, five channels can be added to the processor configuration for growth.

Main Store

The main store of the 3B20S Processor provides a maximum of 8

megabytes of memory. Main store can be equipped with up to 16 memory array (TN14) boards. Each main store array contains 80 64K bit dual in-line packages used for storing data and instructions. Each array has the capability to store 128K 40-bit words. Thirty-two bits (of 40) are data; the remaining eight bits of each word are for error detection. Other circuitry on the memory array board provides for input/output control select decoding and address parity generation.

A main store controller is also provided in the main store unit. The main store controller performs error detection and correction. Hamming code provides correction of all single bit errors and detection of all double errors and detectable multi-bit errors. The state of the memory is constantly being checked during refresh cycles for data parity errors.

In addition to the Hamming code circuitry and data parity checks during a refresh cycle, the controller performs various hardware checks on internal circuitry and bus communications. The main store controller interfaces with the main store bus and the main store arrays.

Cache Store

The cache store unit is a 2K-word, high-speed memory which serves as a local storage area for the most frequently used data words. The primary purpose of cache store is to reduce the real time required to process main store information. The access time to the cache store is approximately 250 nanoseconds (ns) compared to 800 ns for a main store access.

When the central control is ready to read the main store and the cache is available, the central control sends the address to cache store and starts a timer. Cache searches its memory for the information associated with that address. If cache contains the information, it sends a "hit" signal to the central control, indicating that the information is available. When cache signals a "hit", the central control stops the timer and gates the information for that address onto the data bus. If cache does not contain the information for the address, the timer in the central control times out in 250 ns and the address will be sent to the main store. Each word read from the main store is also stored in cache memory, and the cache address is linked to the main store address of that word. If the word is not found in cache, it is fetched from main store to the central control and is also stored in cache. This process of updating cache is referred to as a continuous update. Using this update process, the most often fetched 2K words will usually be found in cache, providing a 70 to 80 percent possibility that the requested information will be in cache. Considering that each successful cache read saves 550 ns in execution time, the cache is a

definite real-time saver.

B. Peripheral Features

All the 3B20S Processor peripherals have direct memory transfer capabilities. The central control builds job blocks for a peripheral device. The peripheral device performs the requested job and notifies the central control upon completion. Peripherals use virtual addresses. Peripherals can initiate direct memory access jobs without central control intervention after the address translation mechanism is set up. Input/output performed in this manner provides parallel operation between the central processor and peripheral devices. From the time the processor completes building a job block until the peripheral devices transfer the desired data into or out of the main memory, the central processor is free to perform other tasks.

Input/Output Interfaces

The primary communication path between the 3B20S Processor and peripheral devices is the direct memory access unit via the central control input/output bus (a high-speed, direct-coupled parallel bus). This bus accesses only a select set of devices called channel units (dual serial channels) and the direct memory access controller.

A dual serial channel interfaces with a maximum of 16 peripheral devices by means of 16 sets of 5-pair private serial data cables. Two serial streams are simultaneously transmitted (hence the name dual serial). This channel can be operated in a word-transfer mode to achieve a transfer time of about 4.5 microseconds per 32-bit word.

The direct memory access unit provides data movement between input/output devices and the main store without the central control handling each individual word transferred. The direct memory access unit consists of a direct memory access controller and a direct memory access input/output bus connected to the dual serial channels. As mentioned previously, each dual serial channel can accommodate as many as 16 peripheral devices. A 3B20S Processor can be equipped with as many as four dual serial channels, providing an interface for as many as 64 direct memory access controlled peripheral devices.

The direct memory access controller serves several major control functions and operates asynchronously from the central control. The direct memory access controller provides:

- An interface with the main store
- An interface with as many as four dual serial channels over the input/output bus
- An interface with the central control over the central control input/output bus

o Tables and sequencing to control data transfers.

The interface between direct memory access dual serial channels and peripheral controllers is provided via the duplex dual serial selector circuit in the input/output processor or disk file controller. The selector receives the dual data streams and formats them for the peripheral device controller.

The 3B20S Processor interrupt structure allows interrupts to be assigned to the input/output in the central control interrupt-source register. The interrupt and service request signals of channels equipped on the central control input/output bus can be connected to different interrupt levels, allowing differentiation between them by priority level. Several channel interrupts can be ORed into a single interrupt bit in the interrupt source register. In this case, interrupt resolution must be obtained by polling each channel or by using interrupt or service request acknowledgment commands. The service request signals for direct memory access channels are not available for connection to the interrupt source register but are handled by the direct memory access controller itself.

Data Link Interface

Various synchronous and asynchronous data link interfaces are supported by the 3B20S Processor. Full duplex Electronics Industries Association (EIA) standard RS232 asynchronous data links are supported at data rates up to 9600 bps. Several synchronous data link interfaces are supported on the 3B20S Processor. BX.25 (Issue 2) level 3 protocol is provided under the UNIX operating system. Level 2 of this protocol is provided be the TN82 (MC4C052) and TN75B peripheral controllers. The TN82 (MC4C052) provides a 56 Kbps full duplex private line data link interface with CCITT V.35 or EIA RS-449 compatible interfaces. The TN75B provides two channels at 4800 bps per channel or 9600 bps when utilizing one channel. Channel 0 of the TN75B controller can be configured as a private line or dial in/out arrangement. Channel 1 is configured for private line use only.

The TN82 (MC4C057) peripheral controller, together with an auxiliary circuit pack UN53, provides four synchronous communication data ports. One of these ports operates at a maximum data rate of 56 Kbps. The other three operate at a lower data rate (below 9.6 Kbps). An additional 128K bytes of memory is provided on the UN53 board allowing a set of peripheral controllers to utilize software for the development of user applications. The UN53 circuit pack is a special pack developed to provide extra memory and bisync protocol capability.

Auto Dial/Answer

The TN75B can be arranged as an automatic dial-out controller for low-speed data links. The TN75B interfaces to an automatic calling unit which is associated with the specific low-speed data

link.

Maintenance Interface

The 3B20S Processor maintenance interface has a maintenance terminal peripheral controller (TN83) which interfaces the maintenance terminal and the local hard copy maintenance printer to the system. The 3B20S Processor maintenance peripheral controller has an emergency action interface (EAI) connection to the central control and a connection to a diagnostic processor. These connections allow the establishment of system configurations and initialization parameters with minimal hardware dependencies.

The maintenance terminal is a 4025BS/001/AF video terminal with split screen capabilities and advanced video options. Four private function keys are provided to invoke special commands to the video terminal peripheral controller and display software. These keys are used to (1) put the maintenance terminal peripheral controller in an emergency action mode, (2) control operations via the emergency action interface ports (i.e., put the central control in an operational mode by forcing it on-line), (3) put the diagnostic processor on-line and (4) run off-line diagnostics, initialize, and boot the system.

The optional diagnostic processor resides on two peripheral controller circuit pack positions of the input/output processor. It interconnects via dual serial channels to the maintenance channel of the central control and to the input/output processor. When put into the diagnostic mode, the diagnostic processor, after diagnosing itself and the input/output processor, uses the magnetic tape on the input/output processor to page in programs to test the central control. Since the diagnostic processor has control of the input/output processor in this mode, it can also use the maintenance terminal and printer to display the diagnostic results..

Moving Head Disk Storage

The 3B20S Processor provides a reliable and flexible mass storage system for programs and data via the moving head disk system. This system provides a means of rapidly storing or retrieving large amounts of data in order to furnish the associated processor with access to a pool of information which cannot be economically stored within the processor main store memory.

The moving head disk storage system is comprised of two main units: (1) the moving head disk transport and (2) the disk file controller. Commercially available moving head disk transports are available in 300-Mbyte sizes. A disk pack for the 300-Mbyte unit has 20 surfaces on 10 disks. Nineteen surfaces are used for data storage, and one is reserved for head positioning information.

The disk file controller provides microprocessor control which interprets and executes commands from the central control to cause information transfers from and to the moving head disk transports. Each disk file controller will support up to eight moving head disk transports.

Disk Power Sequencing Feature

Disk power sequencing limits the current surge on the 208-Vac line during start-up of the disk motor on the drive unit of the 300-Mbyte disk transports. An incoming ground lead is provided by the drive logic to start the disk motor. When the disk motor reaches its maximum speed, a relay contact is closed, passing the ground lead to an output pin and on to the next disk transport. This sequence is repeated until all the disk transports are up to speed. A single transport can be powered down and restarted without having to go through the entire sequence. The disks are ac powered and are not functional during a power outage.

Magnetic Tape System

The KS-22091 magnetic tape system of the 3B20S Processor consists of a tape transport and an associated tape formatter. The tape formatter reads and writes data from and to the tape in a 1600 bits per inch (bpi) single density phase-encoded format. The formatter accepts commands to automatically write an identification code at the beginning-of-tape mark, write an end-of-file mark, properly format blocks of data to be written, and interpret this format on read commands.

The tape transport, via the formatter, reads and writes a standard 9-track tape at a read or write speed of 25 inches per second (ips) and a fast forward or reverse speed of 100 ips. Tape transports are connected to the formatter (when more than one is required) in a daisy chain configuration; i.e., each succeeding tape transport is connected to the preceding unit in a serial manner. During all write operations, the mode of operation is read after each write to detect errors as they occur. The tape transport accepts the following commands:

- o Rewind
- o Forward/reverse
- o Read/write
- o Erase
- o Write end-of-file mark
- o Off-line.

Terminal Compatibility of UNIX 4.1

The operating system for the 3B20S Processor, UNIX release

4.1, includes a number of new terminal features primarily in the form of user-definable options. These options include the following:

- o A wider range of baud rates
- o The ability to ignore breaks
- o The marking of parity errors
- o The selection of fill characters or programmable time for delay
- o The ability to stop and start input device to prevent buffer overflow
- o Intercharacter timing as a function of time or character count
- o User-definable characters for end of line, end of tape, interrupt, erase, kill, and quit
- o The ability to disable "flush" after interrupt or quit.

The implementation of these features is supported by the TN74 asynchronous maintenance terminal peripheral controller firmware.

Line Printer Interface

The line printer interface utilized on the 3B20S Processor is the TN85 peripheral controller. This controller occupies one slot in the 3B20S input/output processor and drives one or two printers from two internally available data streams. A Data Products Long Line-type interface is employed. This interface uses differential drivers operating at transistor-transistor logic levels and can achieve a maximum character transfer rate of 500K characters/second. The actual character throughput is determined by the mechanical characteristics of the line printer employed. Each of the two available line printer ports on the peripheral controller can support printer speeds up to 2000 lines per minute, and the combined throughput of the peripheral controller will not exceed 2000 lines per minute for two printers operating simultaneously. The peripheral controller-to-printer cable is a shielded twisted pair with a maximum cable length of 500 feet. The printers are ac powered and are not functional during a power outage.

The printers used with the TN85 peripheral controller utilize line-matrix, band, or chain technology. Paper is 80 to 132 column pin-feed fan-fold single or multipart forms. All printers support the 96-character ASCII set. Printer hardware diagnostics, if provided, will be in the form of a status display mounted on the printer. Any error condition in the printer raises an error flag in the peripheral controller.

Several commercial line printers have been evaluated and two models have received tentative approval for use with the 3B20S Processor. These printers are as follows:

o Printronix P300 (300 LPM)

o Printronix P600 (600 LPM).

C. Diagnostics

Off-Line Diagnostics

The 3B20S Processor can be diagnosed in a number of ways to provide a comprehensive set of support facilities. Two diagnostic operational modes referred to as on-line and off-line provide the processor with all the self-diagnostic capability normally required for system support. The on-line diagnostic operational mode is based on the UNIX operating system executing in the central control. In the off-line mode, code is executed in a special purpose diagnostic processor. These two modes are mutually exclusive, either of which may be initiated via the emergency action interface (EAI) by a system operator.

In the off-line mode, off-line diagnostics are provided to test the on-line control unit either routinely or on demand. A mini operating system in the diagnostic processor controls the diagnostic processor, the input/output processor, the tape unit, a maintenance terminal, and the associated peripheral controllers. Under control of a maintenance terminal, diagnostic tests are paged from magnetic tape and executed by the diagnostic processor with access to the 3B20S Processor through its maintenance channel. A subset of input and output commands is used to control the testing and to report the results.

When the off-line diagnostic mode is initiated using the emergency action interface, on-board diagnostics are initiated to test the diagnostic processor internal circuitry first. Results of the diagnostic processor diagnosis are passed through the emergency action interface to the maintenance terminal using the process recovery message (PRM) facility. If all tests pass in the resident diagnostic processor diagnosis, the input/output processor is diagnosed again by using the on-board diagnostic processor code.

After diagnosis of the input/output processor, on-board diagnostics of the magnetic tape controller and the maintenance terminal controller are initiated from the diagnostic processor with the process recovery messages indicating the results. The sequence of process recovery messages indicates whether all tests passed or if a failure was encountered, and the diagnostic processor, input/output processor, or controller circuit pack(s) to be replaced. If all tests pass, the diagnostic processor initiates a bootstrap from the magnetic tape of the off-line operating system. Note that since the input/output processor and maintenance terminal controller are essential units in both

operating modes (on-line, off-line) the diagnostics of these units are resident in the diagnostic processor firmware.

On-Line Diagnostics

On-line diagnostics are provided by porting the diagnostic control structure (software) to the UNIX operating system. With this software, any equipment unit not required for basic operation of the system can be logically removed from service. This feature depends on a minimum essential 3B20S Processor configuration consisting of an operative control unit, an input/output processor, and disk community. The interface used to conduct on-line diagnostics include the maintenance terminal, emergency action controls, equipment configuration capabilities, and input/output message formats. Where it is applicable, on-line diagnostics are the preferred diagnostic mode because--the UNIX operating system retains control of the system and a full test data base is available.

D. Power Features

Power Source (ac, dc)

The 3B20S Processor requires source voltages of 117 Vac single phase, 208 Vac three phase, and 208 Vac single phase. The 117 Vac is provided through the service outlets for units such as the video terminal, printer, test equipment, etc. The 208 Vac is the source voltage for a rectifier located in the power conditioning cabinet of the processor system. The rectifier supplies -48 Vdc for system operation. The 208 Vac single phase source is required for each disk drive. All of the above mentioned ac power sources must be supplied by the 3B20S Processor customer. They must be installed and engineered using a single point grounding system and providing a ground window. The ground window is the only point of contact between the components of the single point ground system and other grounded components of the building or of equipment in the vicinity. In other words, it is the window through which the single point ground system sees the rest of the world.

Power Backup

In the 3B20S Processor application, power to the processor, input/output processors, and disk file controllers is distributed from a power conditioning circuitry housed in its own cabinet. This circuitry rectifies commercial ac power and charges batteries within the power conditioning cabinet. In the event of an ac power failure, the batteries provide backup power to the 3B20S Processor for 15 minutes. If outage exceeds this time interval, the system will lose all power.

During an ac power outage, the disk drives will not be functional. As a result, any disk jobs submitted to the disk file controller will fail. To verify the outage, the software reads two power status signals (generated by the power

conditioning circuit) whose states indicate the exact nature of the power fault. In addition to indicating ac power failures, the power status bits carry power conditioning circuit fault information.

To verify the outage of ac power, the software scans two status bits--the battery on discharge bit and the power conditioning fault bit. The battery on discharge bit indicates that some fault is causing the batteries to carry all or part of the -48 volt load. The conditioner fault bit indicates that a fault exists in the power conditioning cabinet circuitry.

The 3B20S Processor software will become aware of a power system fault in one of two ways. First, software periodically scans the battery on discharge bit and the power conditioner bit. If a fault state exists, the appropriate system response is executed. Second, the software checks the bits as a consequence of an ac outage. In addition to indicating ac power failures, the power status bits carry power conditioning circuit fault information.

Auto-Restart Feature

The auto-restart feature allows power to be restored to the central control via signal input to the power switch. This is accomplished by adding a signal input to the ABB1 (power switch) and applying an appropriately timed contact to the closure of this input. The contact closure is generated by the power conditioning circuit as part of its power restoral sequence. Since there are three ABB1s in the basic 3B20S Processor, multiple contact closures are required.

When power is applied to the central control, a maintenance reset function (MRF) signal is generated by circuitry associated with the power switch (ABB1) and power monitoring circuit (TN11). In response to the MRF signal, the central control clears its circuits and forces its microcode into a HALT loop. The central control remains in this halted state until a subsequent MRF signal is generated via the emergency action interface circuit (TN11) to initiate a system boot. Currently, power must be applied to the processor manually by activating switches on each of the ABB1s. This on-site operator intervention is undesirable in the 3B20S Processor application since power shutdowns (due to ac power outages) may be relatively common; therefore, the advent of the auto-restart feature greatly simplifies system restart procedures.

E. System Configuration

The 3B20S Processor equipment units are housed in free-standing cabinets arranged to form a cabinet lineup. A fixed cabinet lineup provides a minimum configuration with conveniently added optional growth units. The minimum configuration consists of:

- o Input/Output Peripheral Control Cabinet
- o Power Conditioning Cabinet
- o Processor Cabinet
- o Disk Controller/Magnetic Tape Cabinet
- o Moving Head Disk Transport
- o Video Terminal
- o Printer.

Options to the minimum configuration include the addition of two input/output processor growth units and an input/output processor basic unit to the input/output peripheral control cabinet. In addition a growth cabinet can be added to the system configuration which contains an input/output processor basic and growth unit and disk file controller. When the second disk file controller is added to the growth cabinet, disks may be added to the system at both ends of the cabinet lineup. The minimum configuration cabinet lineup is approximately 15 feet long. A typical configuration with a growth cabinet and an additional disk is approximately 20 feet long.

One standard size cabinet, 2 feet 2 inches wide, 2 feet deep, and 6 feet high, is used for equipment unit configurations. This cabinet has a hinged door in the front for access to circuit packs and a hinged door in the rear for access to the backplane wiring and cable routing. Within the cabinet, an internal vertical framework provides a mounting surface for front removable units. Cabinets are equipped with fans for cooling where it is required. The base of the cabinet is for air intake; air is exhausted out of the top of the cabinet. All input/output cabling to the system is done in one of two ways. In the first method, the cabling is routed through the base of the cabinet and then routed under the floor (raised floor) to the user cabinet. The second method is through a port in the top of the cabinet via a duct to the ceiling and over the ceiling to the customer equipment. The 208 Vac power for the system is brought into the cabinet through the base or via a port in the rear of the cabinet across the floor to a tel-pole and then into the ceiling.

F. Software/Firmware

UNIX/3B

The UNIX time-sharing operating system is the standard operating system for the 3B20S Processor. The UNIX operating system is a general purpose, multiuser interactive operating system specifically designed to make the user's computing environment simple, efficient, flexible, and productive. The UNIX operating system, functioning with the 3B20S Processor (UNIX/3B), provides a convenient working environment and a

uniform set of tools for computer program development, text processing, and operating support system execution. Various software/firmware features are available with the 3B20S Processor version of the UNIX operating system. The following is a list of some key features available:

- o Communication Line Control--Allows the user to select from a set of communication line methods to use on data link to control unit communications

- o Designated Communications Paths--Allows unrelated processes to send data to each other

- o Process Locking--Allows a process to lock itself in main memory

- o Source Code Control Systems--Consists of a package to aid the maintenance of a software package during development of a set of related programs

- o Text Processing System--Provides an easy to learn and use text processing system which includes:

  (a) A full screen editor formatter

  (b) Interactive spelling checker

  (c) Personalized environment

  (d) Electronic distribution of documents

- o Printer Spooler--Provides for local printing facilities

- o IBM Remote Job Entry--Provides intermachine communications to IBM and other machines

- o BX.25 Protocol--Provides the necessary software to control BX.25 data links

- o Interprocess Communication--Provides three forms of interprocess communication: messages, shared memory, and semaphores

- o Virtual Protocol Machine--Provides the protocol primitives to allow a user to control user-defined bisyncronous data links

- o On-line Diagnostics--Permits diagnostic capability of the periphery of the processor

- o More Extensive Terminal Compatibility--Provides several terminal compatibility features included in the 3B20S Processor version of the UNIX operating system, primarily in the form of user-definable options. These include:

(a) A wider range of baud rates

(b) The ability to ignore breaks

(c) The marking of parity errors

(d) The use of fill character for delay

(e) The ability to stop and start an input device to prevent buffer overflow

(f) Intercharacter timing

(g) User-definable characters for end of line, end of tape, interrupt, and quit

(h) The ability to disable "flush" after interrupt or quit

o UNIX Boot--As a minimum, the following boot features will be supported:

(a) 3B/DGN--Either the 3B20S Processor operating system or the diagnostic processor may be selected (default is 3B20S Processor)

(b) Disk/Tape boot--The operating system may be booted directly from disk or from tape via disk (default is disk)

(c) Device 0/Device 1--Either disk drive 0 or disk drive 1 may be the boot device (default is drive 0)

(d) Boot/Restart--Either the system may be booted or restarted. To restart a system, the boot microcode jumps indirectly through location 4 rather than loading little boot (default is boot). More than one level of restart is supported.

(e) Default OS/name--Either the default UNIX system "UNIX" or an operator-supplied name will be booted (default is default operating system).

UNIX Command Language

The UNIX command language utilizes an extended version of the UNIX shell (command language interpreter) as well as commands designed mainly for use within the shell command files. These commands provide features for the UNIX/3B20S Processor computing facility. For detailed descriptions of the UNIX commands, refer to the UNIX User's Manual (254-301-925WE).

cat - This command reads a file in sequence and writes it on the

standard output; concatenates one or more files and writes them sequentially onto the standard output or in a separate file. Mostly used for unadorned printing of a file or the manipulation of files of data.

pr - Prints the contents of the named file on the standard output.

cp - Copies the contents of one file into another file. This command works on any file regardless of contents.

rm - Removes the entities for one or more files. If any names are linked to the file only the name being removed goes away.

mv - Moves a file. This command is used for renaming files or directories.

ls - For each directory named, this command lists the contents of that directory; for each file named, its name is repeated and any other information requested is listed.

date - Prints the current date and time; can be used to set the current date and time.

tty - Prints the "name" of the user's (your) terminal (i.e., the name of the port to which your terminal is connected).

who - This command lists the login name, terminal name, and login time for each current UNIX user.

pwd - Prints the name of your working (i.e., current) directory.

sort - This command sorts lines of all named files together and writes the results on the standard output. The sorts are conducted in ascending or descending order, alphabetically or on numeric key, on multiple keys located by delimiters, or by position.

diff - This command tells what lines in two files must be changed to bring them into agreement.

grep - This command searches input files and prints all lines in one or more files that match a pattern of the kind used by the editor.

spell - This text processing type command finds spelling errors by collecting uncommon words from a file and looking them up in a large spelling list. Words that neither occur among nor are derivable (by applying certain inflections, prefixes, and or suffixes) from words in the spelling list are also printed.

nroff,troff - These commands are the text processors of the UNIX time-sharing system that format text for typewriter-like terminals and for a graphic system phototypesetter. The text processors can be invoked by a terminal command. These processes

accept lines of text and lines of format control information and format the text into printable paginated documents. These documents have a user-designed style. The control lines begin with a control character followed by a one or two character name that specifies a basic request or the substitution of a user-defined macro in the place of the control line. The text processors can prepare output directly for a variety of terminal types and is capable of utilizing the full resolution of each terminal. The general form of invoking these commands is:

command options files

where "command" is either the fBnrofffP or fBtrofffP command, "options" represents any of a number of option arguments, and "files" represents the list of files containing the document to be formatted. The text processors offer unusual freedom in document styling and layout including:

(1) Arbitrary style headings and footings, including footnotes

(2) Multiple automatic sequence numbering for paragraphs, sections, etc.

(3) Multiple column output

(4) Dynamic font and point size control

(5) Bracket construction

(6) Line drawing functions.

Source Code Control System

The source code control system consists of an integrated set of commands designed to help software development projects control changes to source code and files of text. It provides facilities for storing, updating, and retrieving (by version number or date) all versions of source code modules or documents and for recording who made each software change, when it was made, and why. The source code control system was designed to save most of the source code and documentation control problems that software development projects encounter when customer support, system testing, and development are all proceeding simultaneously.

Some of the main characteristics of the source code control system are as follows:

- o The exact source code or text, as it existed at any point of development or maintenance, can be recreated at any later time.

- o All releases and versions of a source code module or document are stored together, so that common code or text

is stored only once.

- o Releases in production or system-test status can be protected from unauthorized changes.

- o Enough identifying information can be automatically inserted into source code modules to enable one to identify the exact version and release of any such module, given only the corresponding load module or its memory dump.

IBM Remote Job Entry

The UNIX remote job entry feature provides a set of background processes for the submission and retrieval of jobs from an International Business Machines (IBM) host system (e.g., IBM System/360 and /370 host computers). The UNIX operating system communicates with IBMs Job Entry Subsystem by imitating an IBM remote station. At the request of a UNIX user, remote job entry submits a job (via send command); remote job entry gathers the job control statements and sends them to the host processor. When the job is accepted by the host system, a job number is assigned to it and an acknowledgment message is generated. This indicates that the job has been scheduled on the host system. Subsequently upon job completion, remote job entry retrieves from the host processor the resulting output, places it in a convenient UNIX file for later perusal and notifies the user of the outputs arrival.

Text Processing System

The UNIX text processing facilities permit quick and convenient production of many kinds of documentation. Text processing provides a word processing system, an editing system, text formatting systems, a typesetting system, and a spelling and typographical error-detection facility. Text processing facilities include commands for automatically controlling pagination, style of paragraphs, line justification, multicolumn pages, footnote placement, and generation of tables of contents and tables. There are also excellent facilities for formatting and typesetting complex tables and equations.

Display Editor

The display editor is a useful tool for the UNIX user with little or no knowledge of UNIX editing commands. With the display editor, the user can correct, add, or delete characters or lines by controlling the cursor via the arrow keys on the terminals.

The display editor has its own menu, offering choices such as search, forward, backward, top of page, and display. Also the display editor maintains a command bar at the bottom of the screen which constantly flashes the current command mode. The search option prompts for a unique user-supplied pattern and then scrolls forward to the first instance of that pattern. The other commands; forward, backward, top, or display (repeats current 20

lines). Once the screen is full (20 lines), the user may select the edit option or any other directional scroll option. The edit option allows the user to use the cursor positioning keys as long as is necessary to edit the current 20 lines of text. The way to terminate the edit option is to input a carriage return and a control d.

Time-of-Year Clock

A time-of-year clock feature is incorporated into the 3B20S Processor to provide time stamp and process scheduling capabilities for the UNIX system. The time-of-year clock is implemented in software using the 3B20S Processor real-time clock circuit as a reference.

The real-time clock is a 32-bit counter driven by a crystal-controlled oscillator. It has a resolution of 1 millisecond and a range of 49 days. The real-time clock has a read-only register, so it cannot be used as an absolute value. Rather, it serves as a reference to the time-of-year clock software, which compares the current value of the real-time clock to an initial value and uses the difference to generate the absolute time-of-year. The initial value of the time-of-year clock is manually established by a "set date" command to the system terminal. This causes the real-time clock to be captured and associated with the input date value. It should be noted that the time-of-year must be manually re-established any time the processor cabinet is powered down.

7. SOFTWARE DESCRIPTION

SECTION PURPOSE

This section of the system description provides a general description of the UNIX operating system utilized by the 3B20S Processor.

GENERAL

The 3B20S Processor is a general purpose, multiuser processor which utilizes the UNIX operating system. The UNIX operating system provides a convenient working environment and a uniform set of tools for efficient development of computer programs, text and documentation preparation, and other user-defined applications. The UNIX operating system is a disk-oriented system.

User communication with the UNIX operating system is carried on with the aid of the "shell" command language interpreter. Because shell procedures are easy to create and use, programming drudgery is minimized if not eliminated. The shell is the tool that allows users to enhance and build upon the basic UNIX capabilities and adapt the operating system to many, varied, and unique applications without resorting to a compiler. The UNIX operating system typically runs unattended.

OVERVIEW

The UNIX operating system is a general purpose, multiuser, interactive operating system specifically engineered to make the user's, designer's, programmer's, and documenter's processor interface simple, efficient, and flexible, thus resulting in more productive use of the processor. Some of the general features of the operating system include:

- o A flexible, easy-to-use command language that can be "tailored" to specific user needs
- o Flexible document preparation and text processing systems
- o Access to the facilities of other (host) computer systems
- o A high-level programming language conducive to structured programming (C language)
- o Other programming languages, such as Basic
- o Debugging systems

- o A variety of system programming tools (e.g., compiler-compilers)

- o Sophisticated "desk-calculator" packages.

The file system of the operating system consists of a highly-uniform set of directories and files arranged in a tree-like structure. Some of the features of the file structure are:

- o Simple and consistent naming conventions; file names can be fully qualified or relative to any directory in the file system hierarchy.

- o Mountable and demountable file systems and volumes.

- o File linking across directories.

- o Automatic file space allocation and deallocation that is invisible to users.

- o A complete set of flexible directory and file protection modes that allows all combinations of read, write, and execute access, independently for the owner of each file or directory, for a group of users (such as all members of a project), and for all other users. The file protection modes can be set dynamically.

- o Facilities for creating, accessing, moving, and processing files, directories, or sets of these in a simple, uniform, and natural way.

- o Each physical input/output device from interactive terminals to main memory is treated like a file, allowing uniform file and device input and output.

Since the UNIX operating system became operational (1971) many installations have been put into service. Most of the installations are engaged in applications such as computer science education, the preparation and formatting of documents and other textual material, the collection and processing of trouble data from various telecommunications switching machines within the Bell System, and recording and checking telephone service orders. Some installations are used mainly for research in operating systems, languages, computer networks, and other topics in computer science, and also for text processing.

Besides the UNIX operating system proper, some major programs are available to perform the following types of functions:

- o C compiler

- o Text processing

- o Text editing

- o Typesetting
- o Assembler, linking loader, symbolic debugger
- o Phototypesetting and equation setting programs.

There is a host of support service, utility, recreation, and novelty programs. It is worth noting that the system is totally self-supporting. All UNIX software is maintained on the system. This document was generated and formatted by using the UNIX text editor and text formatting programs.

HARDWARE AND SOFTWARE ENVIRONMENT

The UNIX operating system is a computing facility that provides a convenient working environment and a uniform set of tools for computer program development, text processing, and operations support system execution. The 3B20S Processor main store provides for a maximum of 8 megabytes of memory. The main store module can be equipped with up to 16 memory array boards, with each memory board providing 512 kilobytes of memory storage.

A maintenance terminal peripheral controller is used to interface the maintenance terminal, the local hard copy maintenance printer, and the data set connecting to the remote test facility. The peripheral controller has an emergency action interface connection to the central control and a connection to the diagnostic processor. Via these connections or ports, the configurations of the system and initialization parameters can be established with minimal dependency on hardware. Diagnostics can also be initiated and the results displayed on the terminal or printer.

A. Data Links

A high-speed 56 Kbps BX.25 peripheral controller provides a single high-speed data link capable of speeds up to 56 kilobytes per second (Kbps) full duplex. It also provides two modem interfaces for the data link, one is CCITT V.35 compatible, and the other is EIA RE-449 compatible. The link supports a standard input/output interface to user and supervisor processes and can also be used to interface with the shell.

The low-speed BX.25 peripheral controller is a 2-channel synchronous data link peripheral controller that supports a Bell System data communications protocol standard BX.25. The controller supports two full duplex channels at baud rates up to 4800 bits per second per channel or 9600 bits per second when utilizing only one channel. The link supports a standard input/output interface to user and supervisor processes and can also be used to interface with the shell. The low-speed BX.25 peripheral controller can be arranged as an automatic dial-out controller for low-speed data links.

A two circuit pack high-speed peripheral controller can provide four binary synchronous communications channels. One of the channels can operate at a maximum data rate of 56 Kbps. The other three will operate at a lower data rate (below 9.6 Kbps). Besides the four bisyncronous channels, there is an additional 128K bytes of memory provided on the auxiliary circuit pack, which is used for implementing the protocol software.

B. Terminal Capability

A number of terminal capabilities are available as options for the user. These options include:

- o A wide range of baud rates
- o The ability to ignore breaks
- o The marking of parity errors
- o The use of fill characters for delay
- o The ability to start or stop an input device to prevent buffer overflow
- o Intercharacter timing
- o User definable characters for end of line, end of transmission, interrupt, and quit
- o The ability to disable "flush" after an interrupt or a quit.

The 3B20S Processor will support a wide variety of peripheral devices. The equipment configuration and software is easily modified to support additional peripheral devices.

GENERAL FILE SYSTEM DESCRIPTION

The most important role of the UNIX system is to provide a file system. From the point of view of the user, there are three kinds of files: ordinary files, directory files, and special files. Files are named and called (searched for) by sequences (file names) of 14 or fewer characters.

A. Ordinary Files

An ordinary file contains whatever information the user places on it, for example, symbolic or binary (object) programs, and normally resides on a disk. No particular structuring is expected by the UNIX system. A file of text consists simply of a string of characters with lines demarcated by the ASCII new-line character. The structure of files is controlled by the programs that use them, not by the system.

B. Directory Files

Directory files provide the mapping (paths) between the names of files and the files themselves and thus induce a map-like structure on the file system as a whole. Each user has a directory of unique files; the user may also create subdirectories to contain groups of files conveniently treated together. A directory file behaves exactly like an ordinary file except that it cannot be written on by unprivileged programs, so that the system controls the contents of directories. However, anyone with appropriate permission (read, write, and execute) may read or modify a directory just like any other file. The file system maintains several directories for its own use. One of these is the root directory (which may be considered the base or primary directory). Any one of the files in the system can be found by tracing a path through a chain of directories until the desired file is reached. The starting point for such searches is often the root directory. Other system directories contain all the programs (files) provided for general use, i.e., all the commands.

C. Special Files

Special files constitute the most unusual feature of the UNIX file system. Each supported input/output device is associated with at least one special file. Special files are read and written just like ordinary files, but requests to read or write result in activation of the associated device handler rather than the normal file access mechanism. An entry for each special file resides in the device directory "/dev", although a link may be made to one of these files just as it may to an ordinary file. Thus, for example, to write on a magnetic tape (mt) device one may write on the file mt under directory dev thus /dev/mt. Special files exist for each communication line, each disk, each tape drive, and for physical main memory. Of course, the active disks and the memory special files are protected from indiscriminate access by appropriate read and write permissions.

The device driver special files contain data that describe the associated device to the system and enables communication between the system and the device. Device drivers exist for peripheral devices such as terminals, moving head disks, core memory, phototypesetters, magnetic tape drives, communication links, multiplexers, etc. Thus the special files, device drivers, represent particular devices to the system. There is a threefold advantage in treating input/output devices this way: (1) file and device input/output are as similar as possible; (2) file and device names have the same syntax and meaning, so that a program expecting a file name as a parameter can be passed a device name; finally, (3) special files are subject to the same protection mechanism as ordinary or directory files.

D. Pathnames

When the name of a file is specified to the system, it may be

in the form of a pathname, which is a sequence of directory names separated by slashes, "/", and ending in a file name. If the sequence begins with a slash, the search begins in the root directory. The pathname

/alpha/beta/gamma

causes the system to search the root for directory alpha, then to search alpha for beta, finally to find file gamma in beta. Gamma may be an ordinary file, a directory, or a special file. As a limiting case, the name "/" refers to the root itself.

A pathname not starting with "/" causes the system to begin the search in the user's current directory. The simplest kind of pathname, for example, gamma, refers to a file that itself is found in the current directory. The current directory is /alpha/beta.

E. File Protection Mechanism

Although the access (read, write, and execute) protection scheme is quite simple, it has some unusual features. Each user of the system is assigned a unique user identification (ID) number as well as a shared group identification. When a file is created, it is marked with the user ID and group ID of its owner. Also given for new files is a set of protection bits that specify independent read, write, and execute permission for the owner of the file, for other members of the group, and for all other remaining users.

F. Input/Output Calls

The system calls to do input/output are designed to eliminate the differences between the various devices and styles of access. There is no distinction between "random" and "sequential" input/output nor is any logical record size imposed on files by the system. The size of an ordinary file is determined by the number of bytes written on it; no predetermination of the size of a file is necessary or possible.

G. Creating/Rewriting Files

To create a new file or completely rewrite an old one, there is a "create" system call that creates the given file if it does not exist or truncates it to zero length if it does exist: create also opens the new file for writing and, like the open call, returns a file descriptor.

The file system maintains no locks visible to the user, nor is there any restriction on the number of users who may have a file open for reading or writing. Although it is possible for the contents of a file to become scrambled if two users write on it simultaneously, in practice difficulties do not arise. Locks are neither necessary nor sufficient in the system environment to prevent interference between users of the same file. They are

unnecessary because users are not faced with large, single-file data bases maintained by independent processes. There are, however, sufficient internal interlocks to maintain the logical consistency of the file system when two users engage simultaneously in activities such as writing on the same file, creating files in the same directory, or deleting each other's open files.

H. Pointers Within a File

Reading and writing within a file is normally sequential. This means that if a particular byte in the file was the last byte written (or read), the next input/output call implicitly refers to the immediately following byte. For each open file, there is a pointer maintained inside the system that indicates the next byte to be read or written. If n bytes are read or written, the pointer advances by "n" bytes. To do random (direct-access) input/output, it is only necessary to move the read or write pointer to the appropriate location in the file.

USER INTERFACE - THE SHELL

Communication with the operating system (via a terminal) is accomplished with the aid of a command program language called the shell. The shell is a command-line interpreter; it reads lines entered by the user and interprets the lines as requests to execute other programs. Commands may be read from either a terminal or from a file (which allows commands to be stored for later use). In simplest form, a command line consists of the command name followed by arguments (arg, all separated by spaces) to the command. For example:

```
command arg1 arg2   argn
```

The shell splits up the command name and the arguments into separate strings. Then a file with the name "command" is sought; command may be a pathname including the "/" character to specify any file in the system. If command is found, it is brought into memory and executed. The arguments collected by the shell are accessible to the command. When the command is finished, the shell resumes its own execution and indicates its readiness to accept another command by returning a prompt character on the input/output terminal.

If file command cannot be found, the shell generally prefixes a string such as /bin to command and attempts again to find the file. Directory /bin/ contains commands intended to be universally used. (The sequence of directories to be searched may be changed by user as requested.)

A. Standard Input/Output

The description of input/output functions in the file system described above seems to imply that every file used by a program

must be opened or created by the program in order to get a file descriptor for the file. Programs executed by the shell, however, start off with three open files with file descriptors 0, 1, and 2. As such, when a shell program begins, file 1 is open for writing and is best understood as the standard output file. Except under circumstances indicated below, file 1 is the user's terminal. Thus programs that wish to write informative information ordinarily use file descriptor 1. Conversely, file 0 starts off open for reading, and programs that wish to read messages entered by the user read file 0, the standard input file. File descriptor 2 is, like file descriptor 1, ordinarily associated with the terminal output stream. When an output-diversion request with ">" is specified, file descriptor 2 remains attached to the terminal as the standard error, thus commands which may produce diagnostic messages do not silently end up in the output file.

The shell is able to change the standard assignments of these file descriptors from the user's terminal display and keyboard. If one of the arguments to a command is prefixed by '>', file descriptor 1 will, for the duration of the command, refer to the file named after the ">". For example, the shell command:

```
ls
```

ordinarily lists on the terminal the names of all the files in the user's current directory. The shell command:

```
ls >there
```

creates a file named "there" and places the current directory listing in it. Thus the argument ">there" means place output in the file named "there". On the other hand:

```
ed
```

ordinarily enters the text editor, which accepts editor commands from the user via the terminal keyboard. The command

```
ed <script
```

interprets "script" as a file of editor commands; thus "<script" means "take input from file named script."

Although the file name following "<" or ">" appears to be an argument to the command, in fact, it is interpreted p by the shell and is not passed to the command at all. Thus no special coding to handle input/output redirection is needed within each command; the command need merely use the standard file descriptors 0 (read), 1 (write), or 2 (error) where appropriate.

B. Pipelines

An extension of the standard input/output notion is used to direct (or pass) the output from one command to the input of

another without the use of temporary files maintained by the user. A sequence of commands separated by vertical bars causes the shell to execute all the commands and to arrange that the standard output of each command be delivered (piped) to the standard input of the next command in the sequence. Thus in the command line:

```
ls|pr -2|lp
```

the ls command lists the names of the files in the current directory; its output is passed to the print program command pr, which paginates its input with dated headings. (The argument "-2" requests double-column output.) Likewise, the output from pr is input to lp; this command spools its input onto a file for off-line printing.

C. Multitasking

Another feature provided by the shell is relatively straightforward, i.e., a string of commands on a single input line. Commands need not be on different lines; instead commands may be on the same line but must be separated by semicolons (;) thus:

```
ls; ed
```

This command line will first list the contents of the current directory, then enter the editor. Thus two or more commands may be entered on a single line of input, hence multitasking.

D. Background Processes

A related shell feature, the "&", is another useful feature. If a command is followed by "&", the shell will not wait for the command to finish executing before prompting the user for the next input, instead, the shell is ready immediately to accept a new command (even though the previous command has not completed execution). Such a command is executed as a background process. For example:

```
as source > output &
```

causes source to be assembled (as), with source output being passed to output; no matter how long the assembly takes, the shell returns a prompt symbol immediately. When the shell does not wait for the completion of a command, an identification number of the process running that command is printed. This identification number may be used to wait for the completion of the command or to terminate it. The "&" may be used several times in a line or string of commands:

```
as source >output & ls>files &
```

does both the assembly and the listing as a background process. In these examples, output files (output and files) other than the

terminal were provided; if this had not been done, the outputs of the two commands would have been intermingled and meaningless when printed on the terminal.

The shell also allows parentheses in commands such as the above commands. For example:

```
(date;ls) >x&
```

writes the current date and time followed by a list of files in the current directory which is passed to the file x. The shell also returns immediately for another request since the command is executed as a background process.

E. The Shell as a Command; Command Files

The shell is itself a command and may be called recursively. Suppose file "try-out" contains the lines:

```
as source
mv a.out testprog
testprog
```

The mv (move) command causes the file a.out to be renamed (moved to) testprog. The a.out file is the (binary) output of the assembler ready to be executed. Thus if the above three command file lines were typed on the terminal keyboard, source would be assembled, the resulting file a.out renamed testprog, and testprog executed. If all the command lines are in "try-out", the shell command:

```
sh >try-out
```

would cause the shell (sh) to execute the commands sequentially as the lines are listed in file try-out (if the file has execute permission).

The shell has other features including the ability to substitute parameters and to construct argument lists from a specified subset of the file names in a directory. The shell also provides general conditional and looping constructions useful in programming as well as shell variables.

TRAPS

The 3B20S Processor hardware detects a number of program faults, such as references to nonexistent memory, unimplemented instructions, and odd addresses used where an even address is required. Such faults cause the processor to trap to a system routine. Unless other arrangements have been made, an illegal action or fault will cause the system to terminate the process and, under certain circumstances, to write its image on the file "core" in the current directory. A debugger can be used to determine the state of the program at the time of the fault.

Programs that are looping, that produce unwanted output, or about which the user has second thoughts may be halted by the use of the interrupt signal, which is generated by entering the "delete" character or "break" via the terminal. Unless special action has been taken, the interrupt signal causes the program to cease execution without producing a core file. There is also a quit signal used to force a core image file to be produced. Thus programs that loop unexpectedly may be halted and the remains inspected without prearrangement.

The hardware-generated faults and the interrupt and quit signals can, by request, be either ignored or caught by a process. For example, the shell ignores quit signals to prevent a quit from logging out the user. The editor intercepts interrupt signals and returns to its command level. This is useful for stopping long printouts without losing work in progress (the editor manipulates a copy of the file it is editing).

PERSPECTIVE

The success and popularity of the UNIX operating system is largely due to the fact it provides a simple user interface with the processor. The UNIX operating system is designed to make it easy to write, test, and execute programs. The operating system is also easy to maintain. All source programs are available and easily modified on-line.

The user interface to the file system is extremely convenient from a programming standpoint. The lowest possible interface level is designed to eliminate distinctions between the various devices and files and between direct and sequential access. No large "access method" routines are required to insulate the programmer from the system calls. Another convenience is that there are no "control blocks" with a complicated structure partially maintained by and depended on by the file system or other system calls.

NOTES

8. UNIX KERNEL

SECTION PURPOSE

This section of the system description describes the UNIX kernel, which is part of the UNIX operating system.

THE UNIX KERNEL

The UNIX kernel is the software on which everything else depends. It is the only code that cannot be replaced by the user. It maintains the file system, supports system calls, and manages system resources. This code always resides in memory. The following functions supported by the kernel are briefly described:

- o Process control
- o File system
- o Input/output (I/O) system
- o Device drivers.

A. Process Control

The UNIX system schedules and provides an environment for programs. The kernel keeps track of these processes (i.e., programs that are in some state of execution) by the use of several tables. These tables contain data about the status of the process, its whereabouts, etc. Since the kernel decides what actions to take based on tables, UNIX is referred to as a table-driven system.

Process Environment

Programs reside on disk. When a process is invoked, the kernel sets up an executable environment. This environment includes a copy of the process and related data. In order to execute a process, its environment must reside in memory. Also, an entry for the process is placed in the process management table.

Process Management Table

The process management table contains an entry for every process being executed. The entry contains the location of the process environment, priority of the process, and other data about the process. This table and other tables are used by the kernel to maintain a record of processes currently executing.

Priorities and Scheduling

Each process is assigned a priority. The priorities of user processes are assigned by using the most recent ratio of compute time (i.e., central processing unit [CPU] usage) to elapsed time. These times are updated every second for all processes. The priority of a user process that makes little use of the CPU will increase as its CPU-time-to-elapsed-time ratio decreases. Likewise, the priority of a process that makes heavy use of the CPU will decrease. By this scheme, priorities can be changed during execution in order to allow all user processes equal time. Scheduling processes is straightforward. The process with the highest priority is executed first.

Swapping

Memory is sometimes not large enough to contain all the environments of processes. When memory is not available, an inactive process (i.e., its environment) is copied to disk and another process that is ready to execute is copied from disk. This is referred to as swapping. The kernel maintains a list of available memory space and list of available swap space (on disk) in order to perform swapping.

In addition to the control of processes, the kernel provides means by which processes can communicate with each other. These mechanisms include

- o Pipes
- o Messages
- o Shared memory
- o Semaphores.

These can be used to send the output of one process to another. The receiving process accepts this information as input.

B. File System

The UNIX file system provides a means for storing and organizing data. All data stored on the UNIX system resides somewhere in the file system. Data is stored as a sequence of bytes in a structure referred to as a "file". There are three types of files:

- o Ordinary files
- o Special files

- o Directory files.

Ordinary Files

This type of file is used to store data. The data is stored as a string of bytes. No structure internal to the file is required by the system. Each file has the potential to contain up to about 1,000,000,000 bytes of data.

Special Files

These files are not used to store data. A special file serves as an interface to a specific hardware entity. For example, communication with a disk is via a special file. There exists a special file for every disk unit, tape unit, terminal, etc., attached to the system.

Directory Files

A directory provides a means for organizing ordinary and special files. The system is responsible for keeping the directories up-to-date. A directory contains the names of files that reside "under" it and internal references to those files.

Owner Identification

Each user is assigned a unique identification number. Also, the UNIX operating system supports a "group" concept. That is, several users that are working on the same project (or in the same organization, etc.) can all be assigned to the same group identification number. When a user creates a file, the file is marked with that user's identification number and the group's identification number. Thus, the user that created the file is the "owner" of the file.

File Protection

The user and group identification numbers are used by the file system to permit and deny access to files. The protection of a file is referred to as the "mode" of the file. The mode of a file can only be changed by its owner or the System Administrator. The file mode can be independently defined for:

- o The owner
- o The group to which the owner belongs
- o All others.

Protection of a file is based on three types of permissions:

- o Read
- o Write

o Execution for programs or search for directories.

For example, the owner of a file can be given read and write permission. At the same time other members of the group can be given read permission, but not write permission. Also, access by others that are not a member of the owner's group can be denied. Any combination of permissions is allowed. Also, the system provides a means for encrypting the contents of a file by its owner. Before a file is encrypted, the system requests a password that is to be used to decrypt the file. Permission to read and/or write an encrypted file is denied (regardless of the mode of the file) until the correct password is entered. The encrypting routine is based on the Data Encryption Standard. This is the standard set by the National Security Agency of the United States Government.

File System Structure

The UNIX file system is organized into a hierarchical tree-like structure by the use of directories. Ordinary and special files can be placed "under" any directory. This structure is very similar to an upside-down tree. Directories can be thought of as branches. Ordinary and special files can be thought of as leaves. The starting point of the entire file system is a directory referred to as "root", which has the name "/". Figure 8.1 shows a representation of a file system. The trunk of the tree represents the directory root. Three branches (directories) have been given the names "BWK", "KT", "DMR". Each one of these branches is shown supporting a leaf (i.e., ordinary or special file). Typically, the file system is represented by a line drawing as shown in Fig. 8.2. Figure 8.2 is similar to Fig. 8.1 but shows a more complicated structure. With the exception of portions of the file system structure required by the system, the user can structure the file system as need be. Thus, the UNIX file system is extremely flexible.

Mountable File Systems

Several file systems can be tied into the system at the same time. All these file systems do not have to reside on the same disk drive. Another file system can be mounted as a directory of the main file system. The main file system of the UNIX operating system is the file system that contains the directory root (i.e., "/"). On Fig. 8.1 for example, another branch could be added to root. This new branch could be as large as the other main branches. Mountable file systems can be unmounted and mounted in another place as need be.

C. Input/Output System

The input/output (I/O) system provides a means by which data can be transferred between processes and peripheral devices. The I/O system includes special processes that provide direct control over these devices. These special processes are referred to as device drivers. The user communicates with the peripheral

devices by the use of system calls. The libraries and system calls direct the appropriate device driver(s) to perform the requested action.

The I/O system is divided into two systems. These are:

o Block (structured) I/O system

o Character (unstructured) I/O system.

Peripheral devices are identified by a major device number, a minor device number, and a class (block or character). For each class of device, there is a data table. Each table entry represents a device. An entry contains functions used to perform I/O with that device. When a particular device is to be accessed, the major device number is used to locate the table entry for that device. The minor device number is sent to the device driver when called. The device driver uses this number to access one of several identical devices.

Block I/O System

The model for the block I/O system consists of storage (usually disk) that is divided into blocks of 512 bytes each. Blocks can be addressed randomly. Block devices are accessed through a layer of buffering software. That is, a list of buffers are maintained by the system to temporarily store information. On a write to a device, data is stored in the buffers until they are all filled. At that time, data is transferred to the actual device via the device driver. On a read, any data that was transferred from the device to the buffers is made available to the requesting software. When all buffered input has been accessed by the requesting software, the device driver fills up the buffers again with information from the device. This scheme results in a large reduction of physical communication with the device.

Character I/O System

The character I/O system consists of all devices that do not conform to the block I/O model. This includes magnetic tape, communication lines, terminals, line printers, memory, etc.

Each character device driver maintains a character list. Common code handles the lists. One routine places a character on the list. Another routine gets a character from the list. Storage for the lists comes from a common pool of storage maintained by the system.

On output to a typical character device, the system automatically synchronizes the common code so that the list never overflows.

Input from a character class device is handled in a similar manner. The device driver fills a list with characters from the

hardware device and passes it to the user.

Output to terminals is handled in the manner previously described. But two lists are maintained for input from the terminal. One list maintains the sequence of characters as entered on the terminal. The system processes the characters from the first list and builds the second list. Software can accept input from either list. Thus, the UNIX operating system provides the ability to receive direct input or system processed input.

D. Device Drivers

Device drivers are interfaces that provide the necessary actions to communicate with different hardware devices. A device driver exists for each hardware device that requires a specific protocol. On the UNIX system, there are device drivers for the following types of devices:

- Phototypesetters
- Moving head disks
- Virtual protocol machines (VPMs)
- Magnetic tape drives
- Line printers
- Memory
- Terminals
- Asynchronous multiplexers.

Access to device drivers is through special files. Thus, a special file represents devices and their drivers on the system. For example, information sent to a specific terminal is written to a special file that represents the terminal. The associated device driver handles the protocol with the terminal to allow transmission.

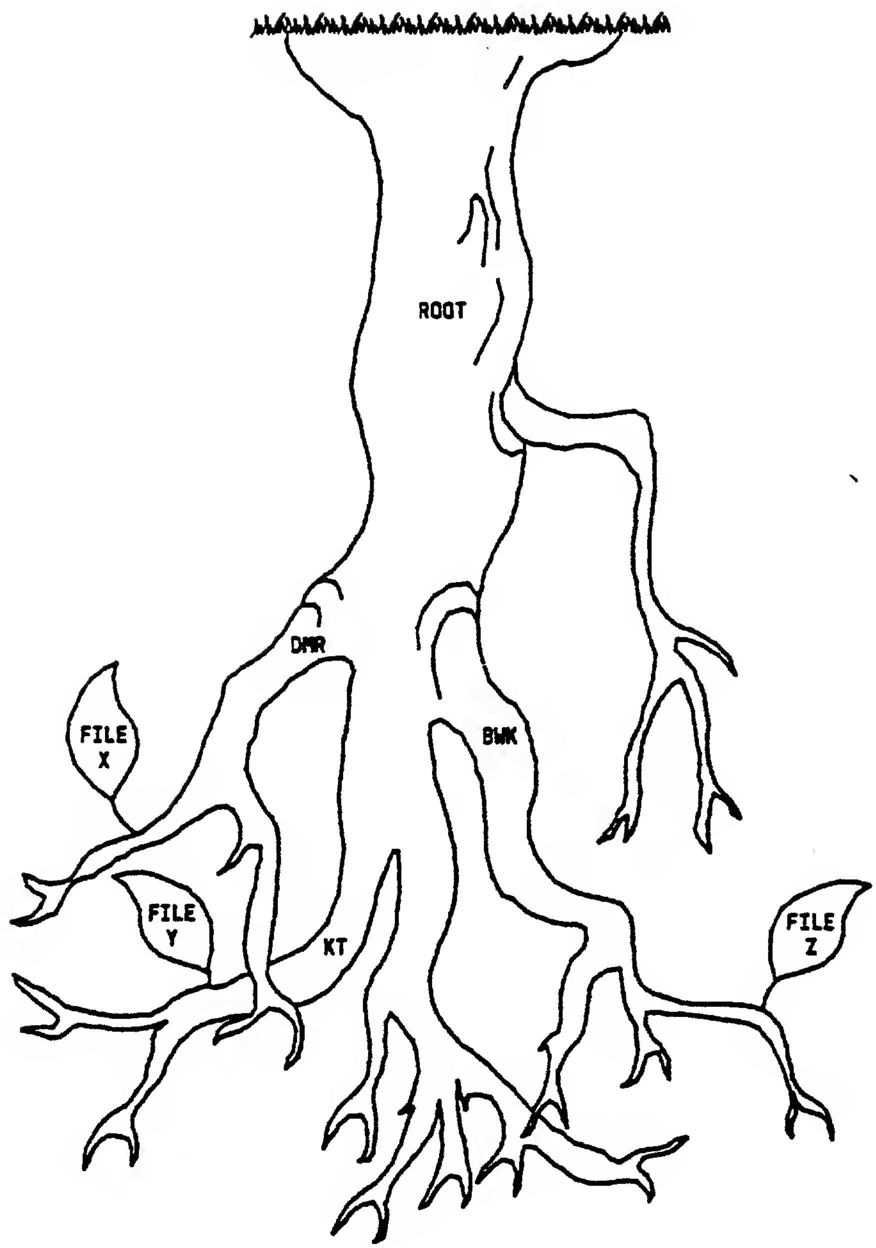

Fig. 8.1--Representation of UNIX File System

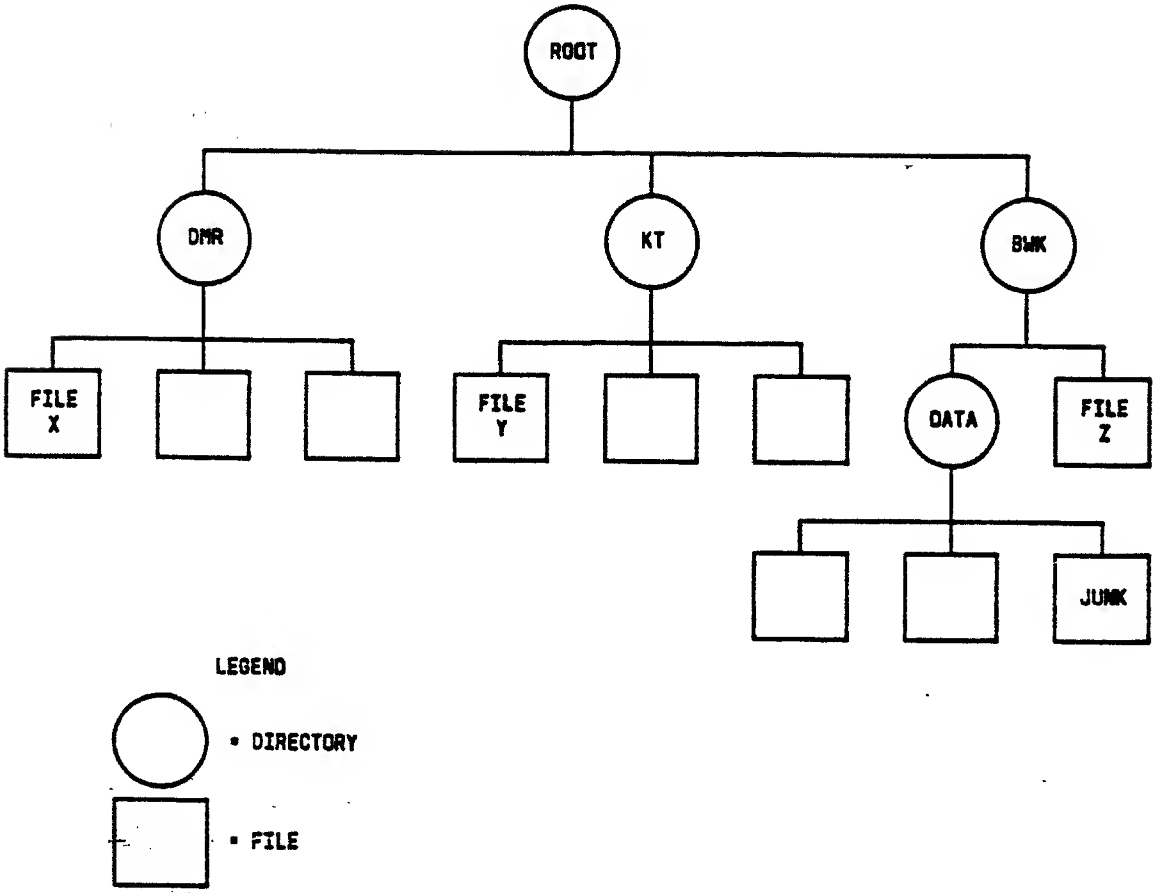

Fig. 8.2--Structure of File System

COMMENT/ERROR REPORTING FORM

3B20S PROCESSOR DOCUMENTATION

Use this form to report any comments, errors, or suggestions for improving the contents of the documents listed below. Place a check beside the document number for which the comment suggestion, or error applies.

___	254-301-910WE	System Index and Description
___	254-301-920WE	UNIX User's Guide
___	254-301-930WE	UNIX Administration Guide
___	254-301-940WE	UNIX Operations Guide
___	254-301-950WE	System Maintenance Guide

Comment: ____________________

RETURN TO:

Western Electric Company
2400 Reynolda Road
Winston Salem, N. C.
27106 Dept 4245

SUBMITTED BY:

Name ____________________
Address ____________________

Phone ____________________